

2018

## Agent-based resource deployment strategies in emergency management

Jihang Zhang  
*University of Wollongong*

Follow this and additional works at: <https://ro.uow.edu.au/theses1>

### University of Wollongong

#### Copyright Warning

You may print or download ONE copy of this document for the purpose of your own research or study. The University does not authorise you to copy, communicate or otherwise make available electronically to any other person any copyright material contained on this site.

You are reminded of the following: This work is copyright. Apart from any use permitted under the Copyright Act 1968, no part of this work may be reproduced by any process, nor may any other exclusive right be exercised, without the permission of the author. Copyright owners are entitled to take legal action against persons who infringe their copyright. A reproduction of material that is protected by copyright may be a copyright infringement. A court may impose penalties and award damages in relation to offences and infringements relating to copyright material.

Higher penalties may apply, and higher damages may be awarded, for offences and infringements involving the conversion of material into digital or electronic form.

Unless otherwise indicated, the views expressed in this thesis are those of the author and do not necessarily represent the views of the University of Wollongong.

---

### Recommended Citation

Zhang, Jihang, Agent-based resource deployment strategies in emergency management, Doctor of Philosophy thesis, School of Computing and Information Technology, University of Wollongong, 2018.  
<https://ro.uow.edu.au/theses1/315>

Research Online is the open access institutional repository for the University of Wollongong. For further information contact the UOW Library: [research-pubs@uow.edu.au](mailto:research-pubs@uow.edu.au)



# Agent-Based Resource Deployment Strategies in Emergency Management

Jihang Zhang

*This thesis is presented as part of the requirements for the conferral of the degree:*

Doctor of Philosophy

Supervisors:

Prof. Minjie Zhang

Dr. Fenghui Ren

Dr. Jiakun Liu

The University of Wollongong

School of Computing and Information Technology

March, 2018

This work © copyright by Jihang Zhang, 2018. All Rights Reserved.

No part of this work may be reproduced, stored in a retrieval system, transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior permission of the author or the University of Wollongong.

This research has been conducted with the support of an Australian Government Research Training Program Scholarship.

# Declaration

I, *Jihang Zhang*, declare that this thesis is submitted in partial fulfilment of the requirements for the conferral of the degree *Doctor of Philosophy*, from the University of Wollongong, is wholly my own work unless otherwise referenced or acknowledged. This document has not been submitted for qualifications at any other academic institution.

---

**Jihang Zhang**

July 24, 2018



# Abstract

In urban areas with high-density population, emergency incidents such as vehicle accidents, medical emergencies and house fires could occur randomly at any locations with uncertain information. These incidents are usually hard or impossible to be predicted in advance and require emergency departments or services to make fast responses to deploy resources for conducting rescue operations after receiving emergency calls and modify the rescue operations dynamically in real time situations based on the changes of incidents and their surrounding environments, thus to minimise the potential losses associated with human lives and public properties.

With the rapid increase of emergency service demands in metropolitan regions around the world, human operators have to handle the emergency responses for concurrent incidents with different contents, severities and resource requirements, which further complicate the resource deployment processes. Consequently, these concurrent emergency incidents could easily overwhelm traditional emergency response systems and cause problems such as inefficiency of rescue resource management, resource deployment delay, resource contention etc.

This thesis mainly aims at researching emergency resource deployment approaches that are capable of automatically coordinating multiple emergency incidents and intelligently providing efficient and flexible resource deployment plans based on resources attributes, incidents' attributes and the variations of environments. The major contributions of this thesis are summarised as follows.

- A comprehensive mathematical model is proposed to represent and model the complicated relationships among emergency incidents, rescue resources, resource deployment objectives, constraints and environmental information.

This model is designed with high-level abstraction terms, so it can be used to handle different types of emergency incidents that require multiple resources from different emergency services.

- An agent-based framework is developed to automatically deploy resource for a single emergency incident. This framework is designed to generate the optimal resource deployment plan based on resource money expenditure and deployment time without the global information.
- An agent-based emergency resource deployment approach is proposed to handle concurrent emergency incidents. This approach contains an effective solution for addressing frequently occurred resource contention problems during the process of multi-incident resource deployment.
- Three dynamic resource deployment mechanisms are proposed to enable dynamically resource deployment during an emergency response process. These mechanisms are designed to dynamically adjust resource deployment plans to handle problems caused by incident variation, task variation and execution variation during an emergency response process.
- A smart emergency response system is developed. This system is designed with an extendable architecture to fit the practical deployment requirements in real-world environments and is capable of generating resource deployment plans based on emergency response codes in different regions and countries for real-life emergency situations.

In summary, this thesis includes both mathematical theories and applications of emergency resource deployment approaches. The experiment results and analysis demonstrate the efficiency and effectiveness of the proposed work in this thesis.

# Acknowledgments

A 3.5-year PhD is a long journey to walk through. I might not be able to reach the destination without the countless helps from various people.

First of all, I would like to express the deepest appreciation to my supervisors, Professor Minjie Zhang, Dr. Fenghui Ren and Dr. Jiakun Liu, for providing me with expert, sincere and valuable guidance and assistance for not only my PhD study but also my daily life over the last 3.5 years. Their advice on both my academic research as well as on my career has been priceless. Without their immense support and persistent help, this thesis would not have been possible to be completed.

In addition, I would like to appreciate the School of Computing and Information Technology for providing necessary technical support during my PhD study. I also gratefully acknowledge the University of Wollongong by providing me with two scholarships, namely the University Postgraduate Award and the International Postgraduate Tuition Award, for my PhD studies.

Finally, I would like to thank my parents, Ming Cheng and Fan Zhang, for their unceasing encouragement, support and love during my study in Australia. I cannot even make one step forward without them. My sincere thanks also go to all of my friends who incite me to strive towards my goal.

# Publications

The following is a list of my research papers, which have been published or are currently under review, during my PhD study.

## Refereed Journal Article:

- **Zhang, Jihang**, Minjie Zhang, Fenghui Ren, and Jiakun Liu. “An Innovation Approach for Optimal Resource Allocation in Emergency Management.” *IEEE Transactions on Computers*, 10.1109/TC.2016.2601331, 2016.

## Refereed Conference Papers:

- **Zhang, Jihang**, Minjie Zhang and Fenghui Ren, “A Dynamic Resource Allocation Approach for Concurrent Emergency Events in Metropolitan Regions.” *In Proceedings of the 16th International Conference on Autonomous Agents & Multiagent Systems*, pp. 1775-1777, San Paulo, Brazil, 2017. (**AAMAS is the top conference in agent and multi-agent research.**)
- **Zhang, Jihang**, Minjie Zhang, Fenghui Ren and Jiakun Liu. “Enable Efficient Resource Deployment in Multiple Concurrent Emergency Events Through a Decentralised MAS.” *In Proceedings of the 29th Australasian Joint Conference on Artificial Intelligence*, Lecture Notes in Artificial Intelligence, pp. 56-68. Springer, 2016. (**The Best Student Paper Award**)
- **Zhang, Jihang**, Minjie Zhang, Fenghui Ren and Jiakun Liu. “A Decentralised Multi-Agent System for Emergency Resource Allocation in Metropolitan Regions.” *In Proceedings of the 15th International Conference on Autonomous Agents & Multiagent Systems*, pp. 1482-1484. Singapore, 2016. (**The Best Demo Nomination in AAMAS 2016**)

**Papers Under Review:**

- **Zhang, Jihang**, Minjie Zhang and Fenghui Ren, “An Agent-Based Dynamic Resource Deployment Approach for Improving Emergency Responses to Concurrent Incidents”, *IEEE Transactions on Emerging Topics in Computational Intelligence*, (Submitted in January. 2018, under review).
- **Zhang, Jihang**, Minjie Zhang and Fenghui Ren, “Providing Smart Emergency Responses for Emergency Incidents Through Agent-Based Resource Deployment”, *the 27th International Joint Conference on Artificial Intelligence*, (Submitted in January. 2018, under review).

# Contents

<b>Abstract</b>	<b>iv</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Background Knowledge of Resource Deployment in Emergency Management . . . . .	2
1.1.1 Emergency Resource Deployment Process . . . . .	2
1.1.2 Present Situation of Emergency Response Demands . . . . .	3
1.2 Agent-Based Emergency Resource Deployment . . . . .	4
1.2.1 Agent and Multi-Agent Technologies . . . . .	4
1.2.2 Advantages of Using Agent and Multi-Agent Technologies for Emergency Resource Deployment . . . . .	5
1.3 Research Challenges of Resource Deployment in Emergency Management . . . . .	7
1.4 Research Issues and Objectives of This Thesis . . . . .	10
1.5 Research Contributions of This Thesis . . . . .	14
1.6 Organisation of This Thesis . . . . .	16
<b>2 Literature Review</b>	<b>18</b>
2.1 Emergency Resource Deployment Modelling . . . . .	19
2.1.1 Modelling Resource Deployment Based on Incident Types . . . . .	19
2.1.2 Modelling Resource Deployment Based on Objective Optimisations . . . . .	24
2.2 Emergency Resource Deployment for Concurrent Incidents . . . . .	29
2.3 Dynamic Emergency Resource Deployment Approaches . . . . .	33

2.4	Emergency Management Systems and Simulators . . . . .	34
2.5	Summary . . . . .	40
<b>3</b>	<b>An Innovative Approach for Optimal Resource Deployment to A Single Emergency Incident</b>	<b>41</b>
3.1	Definitions and Problem Description . . . . .	42
3.1.1	Definitions of Domain Knowledge . . . . .	42
3.1.2	Problem Description . . . . .	46
3.2	Theoretical Foundation of the Domain Transportation for the Opti- mal Resource Deployment . . . . .	48
3.3	Agent-based Decentralised Resource Deployment . . . . .	51
3.3.1	Definitions of Agents and Implementation . . . . .	51
3.3.2	Resource Deployment Framework and Process . . . . .	53
3.4	Experiments . . . . .	57
3.4.1	Experimental Setting . . . . .	58
3.4.2	Experimental Results and Analysis . . . . .	61
3.5	Case Study . . . . .	64
3.5.1	Case Study Setting . . . . .	64
3.5.2	Case Study Results and Analysis . . . . .	68
3.6	Summary . . . . .	72
<b>4</b>	<b>An Agent-Based Dynamic Resource Deployment Approach for Con- current Incidents</b>	<b>73</b>
4.1	Definitions and Problem Description . . . . .	75
4.1.1	Definitions of Domain Knowledge . . . . .	75
4.1.2	Problem Description . . . . .	77
4.2	Agent-based Resource Deployment for Concurrent Emergency Incidents	79
4.3	Dynamic Resource Deployment Mechanisms . . . . .	83
4.3.1	A Resource Deployment Mechanism for Incident Variation . .	83
4.3.2	A Resource Deployment Mechanism for Task Variation . . . .	84

4.3.3	A Resource Deployment Mechanism for Execution Variation . . . . .	87
4.4	Experiments and Evaluation . . . . .	91
4.4.1	Experimental Setting . . . . .	91
4.4.2	Result Analysis and Discussions . . . . .	93
4.5	Summary . . . . .	98
<b>5</b>	<b>A Smart Emergency Response System in Real-World Environments</b>	<b>99</b>
5.1	Problem Description . . . . .	100
5.2	A Resource Deployment Model for Resource Usage Rationalisation and Definitions . . . . .	103
5.3	Design Detail of the Agent-Based Smart Emergency Response System for Multiple Incidents in Dynamic Environments . . . . .	106
5.4	System Deployment in Real-World Environments . . . . .	111
5.5	Experiments . . . . .	113
5.5.1	Experimental Settings . . . . .	113
5.5.2	Experimental Results and Analysis . . . . .	115
5.6	Summary . . . . .	117
<b>6</b>	<b>Conclusion and Future Work</b>	<b>119</b>
6.1	Contributions of This Thesis . . . . .	120
6.2	Future Work . . . . .	122
	<b>Bibliography</b>	<b>124</b>
<b>A</b>	<b>An Agent-Based Emergency Resource Deployment Simulator</b>	<b>136</b>
A.1	Implementation Features of the Simulator . . . . .	136
A.2	Simulator Demonstration . . . . .	138



# Chapter 1

## Introduction

*Emergency management* refers to the creation of plans to avoid the risk or decrease the impact caused by emergency incidents [DH91; Pet85]. Generally, the emergency incidents can be caused by natural disasters or the interaction of people with environments. More precisely, natural disasters are those large-scale incidents that occur as a result of natural processes, such as tornados, freezes, blizzards, earthquakes, hurricanes, etc. Incidents caused by human actions are events that happen in small-scale environments (i.e. metropolitan regions), such as urban fire, industrial accidents, communication failures, acts of terrorism, etc. [HBC13].

Traditional emergency management usually consists of four phases, which are prevention, preparedness, response and recover [Haw+12; Pet85; Rub12]. For large-scale incidents caused by natural disasters, the property damage and the number of fatalities can be effectively reduced during the preparedness phrase by preparing rescue plans, procedures and equipment in advance. However, for incidents caused by human actions, damage mitigation is heavily rely on the emergency response, which refers to the plans for deployment and coordination of rescue resources to mitigate the damage caused by the emergency incidents [MBT13; Wex+14].

In recent years, with the increasing of emergency response demands globally, emergency departments in the majority of cities suffer significantly pressure of handling the resource deployment processes for emergency incidents [Abo+15]. This PhD study primarily focuses on the research of resource deployment strategies for emergency incidents that occur in metropolitan regions by using agent and multi-agent

technologies [WJ95].

## 1.1 Background Knowledge of Resource Deployment in Emergency Management

### 1.1.1 Emergency Resource Deployment Process

In emergency management, emergency incidents that occur around urban areas are usually handled by emergency response processes.

*Emergency response* is the organising, coordinating, and deploying available resources to bring emergency incidents under control, in which deploying appropriate resources for rescue operations is the most crucial process for saving people's lives and reduce finance losses [Tur+04; Tur02].

The *resource deployment process* for the emergency response to an incident usually consists two phases, which are the generation of a resource deployment plan and the execution of the resource deployment plan [ZA08]. The detail of the two phases is described as follows.

- **Phase 1: Resource Deployment Plan Generation.**

In this phase, emergency operators need to manually contact multiple emergency departments to select appropriate rescue resources for emergency incidents from different facilities [Swe94], such as police stations, hospitals, fire station, etc. The resource selection criteria are usually based on the response codes of the emergency incidents, which represent the propriety or severity of incidents. For instance, in East Anglia, United Kingdom, incidents are classified by grade-based emergency service codes, with each incident being labelled by A, B, C or D [18c]. Normally, incidents with different emergency response codes might have different objectives or constraints that need to meet during the resource dispatching (deployment) process, such as resource money expenditure, resource deployment fairness, resource deployment time, task completing number, etc [HLH16; Su+14; SL16].

- **Phase 2: Resource Deployment Plan Execution.**

In this phase, rescue resources, selected for the emergency responses to incidents, need to choose appropriate routes to arrive incident locations and emergency operators need to constantly track dispatched resources to ensure that the resources can be delivered with the predefined time limits of emergency incidents [Chi+08]. If certain resources cannot arrive the incident location in time due to problems such as resource contention [YH95] or traffic jam, emergency operators need to immediately coordinate replace resources for the incidents to guarantee rescue operations can still be conducted on time.

### 1.1.2 Present Situation of Emergency Response Demands

In recent decades, with the rapid increase of population in metropolitan regions, the frequency of occurrence of emergency incidents has been increasing significantly, which results in the high demands of emergency responses. For example, according to the statistical research conducted by Aboagye-Sarfo et al. [Abo+16], in the metropolitan regions of Western Australia, the number of emergency transportation demands for medical incidents rose from 94,369 in 2005 to 153,374 in 2014, and a CAGR (Compound Annual Growth Rate) increase of 5.5% compared to a population growth of 2.9% annually during the same period. It is predicted that the emergency transportation demands will increase by another 30% in 2020.

In recent years, innovations in wireless communications, networking and information technologies have brought significant benefits for emergency response. For example, the research of auto emergency detection and analysis systems such as eCall [18d], AIDER (Accident Information and Driver Emergency Rescue) [18a] and E-health care [Pau+09] play an essential role in decreasing emergency response time. Furthermore, the development of computer-aided dispatch (CAD) systems [Saa06; Wha95] also significantly improved the effectiveness of resource management process.

Nevertheless, the majority of emergency response services around the world are

still relying on human operators to manually select resources and deploy the resources to emergency incidents [LP00; Cla13], which is highly inefficient. This is because, in a metropolitan region, large numbers of rescue resources from different emergency departments or service providers are distributed at different locations. Operators of emergency incidents usually have difficulties to efficiently find out the optimal resource deployment for an emergency incident due to a large number of possibilities. For example, in Sydney region, there are 1500 vehicles, which provide ambulance services with more than 4000 ambulance operators; about 338 fire stations with over 6500 firefighters; and over 6500 police officers working in about 300 police stations [18j]. Selecting resources from such a large resource pool could easily and possibly lead to possible human mistakes due to the evaluation of many factors in a very short time. It is no doubt that public emergency departments suffer significantly from the pressure about how to efficiently and effectively deploy rescue resources for emergency incidents to reduce casualties.

## 1.2 Agent-Based Emergency Resource Deployment

### 1.2.1 Agent and Multi-Agent Technologies

Recently, agent and multi-agent systems have been becoming promising technologies for resource deployment in emergency response [Fie06; GKR11; LIB08; Mar+09; BMM11]. Generally, an *agent* refers to a software system with the capabilities of perceiving, reasoning, acting and communicating with others to achieve its goals. An agent usually possess four properties [Woo97; WJ95; ZO04], which are listed as follows.

1. **Autonomy:** an agent has the ability to control its behaviour and internal state without direct external intervention.
2. **Re-activeness:** an agent has the ability to perceive the environment changes and adjust its behaviour accordingly when it tries to achieve its goals.

3. **Pro-activeness:** an agent is expected to have built-in goals and tries to accomplish them by planning, reasoning and taking actions.
4. **Social ability:** an agent has the ability to interact, cooperate, negotiate and even compete with other agents to achieve its objectives.

A *multi-agent system* (MAS) refers to a computerised system that consists of multiple tightly-coupled or loosely-coupled autonomous agents, which usually need to communicate with each other to accomplish common objectives in a shared environment [WJ95; Woo97; Fer99]. The communication between agents can vary from simple information changing to the action requests, coordination and negotiation.

Over the years, agent and multi-agent technologies have shown great potentials to handle critical needs in high-speed, task-critical, decentralised applications and systems where uncertainty, dynamic environments, mutual interdependencies and sophisticated control play a significant role [BP12].

### 1.2.2 Advantages of Using Agent and Multi-Agent Technologies for Emergency Resource Deployment

Agents and multi-agent technologies offer potential solutions for resource deployment of emergency incidents due to their abilities of autonomous reasoning, collective decision making and group collaboration [ZO04]. More precisely, emergency incidents in metropolitan regions usually require multiple resources that belong to different emergency departments and distributed over a large area. In order to efficiently deploy rescue resources under these emergency incidents' limited time constraints, multiple emergency departments need to cooperate together and each of them might responsible for different resource deployment tasks [Che+08], which is a complicated process. The advantages of using agent and multi-agent systems as promising technologies for solving emergency resource deployment problem are described as follows.

**1. Agents have the ability to perform dynamic actions in open environments.**

Agents' re-activeness properties allow them to dynamically adjust their resource deployment plans and strategies on real-time based on the perception of their surrounding areas, which is critical for conducting emergency response operations in open environments where incidents' contents and resources availability may frequently change [GF06].

**2. Agents have the ability to conduct distributed decision making.**

In multi-agent systems, agents can cooperate with each other in decentralised networks to gather, process, verify and analyze multiple levels of information through self-organisation mechanism, which is extremely important for deploying distributed rescue resources from different emergency departments for concurrent emergency incidents [Ste90].

**3. Multi-agent systems can be easily used for modelling the emergency resource deployment problem.**

By using multi-agent collaboration, coordination and communication mechanisms, the relationships and actions among multiple emergency incidents, rescue resources, emergency departments and environments can be modelled by agents with suitable roles and built-in goals [Muk+17]. Besides, multi-agent systems can be integrated with human-like decision-making strategies [NSR00], such as Recognition-Primed-Decision or Belief-Desire-Intention models, which can significantly increase the practicability of agent-based resource deployment approaches for emergency response in real-world environments. Some of the examples include FireGrid [Ber+05], RoboCup Rescue [Kit+99], DEFACTO [MST06], DrillSim [Bal+06] and ALADDIN [Ada+08].

### 1.3 Research Challenges of Resource Deployment in Emergency Management

Although agent and multi-agent technologies are highly suitable for resource deployment in emergency management, there are still many potential research challenges need to be solved before they can be used in real-life situations. This section outlines the four major research challenges of resource deployment for emergency incidents as follows.

- **Challenge 1: Unpredictability**

Emergency incidents are hard to be predicated in open environments such as city urban areas [HBC13]. They could randomly occur at any possible location at any time, which makes resources pre-deployment become almost impossible. These incidents usually have strict time limits for emergency departments to respond and conduct emergency rescue operations, which further increases the difficulty of finding the most appropriate resource deployment plans for these incidents.

- **Challenge 2: Complexity**

Resource deployment for emergency incidents usually involves the information of environment, emergency incidents, rescue resources, resource deployment objectives, constraints, etc [Muk+17; HLH16]. Processing all this information simultaneously could be a very complicated process [Sno02].

In detail, the resource deployment process of an emergency incident could consist of multiple requirements. Each of these requirements might have different constraints in terms of resource functionality, resource deployment time, resource money expenditure, resource suitability, etc. Therefore, handling the resource deployment of the incident would require multiple emergency departments or services to collaborate with each other to propose effective and efficient resource deployment plans. For example, a house fire labelled with a life-threatening response code might require multiple ambulances and fire

engines with different constraints of their deployment time. For such an emergency incident, it is important to ensure the fire engines can arrive the incident location first to put out the fire, so the ambulance team can conduct rescue operations in time when they have arrived lately.

Furthermore, due to the large amounts of emergency response demands in metropolitan regions, emergency operators and rescue resource dispatchers might need to process emergency responses for concurrent incidents that have different contents, response codes, and resource deployment requirements. It is no doubt that the significant complexity of the simultaneous resource management (i.e. resource deployment, coordination and real-time tracking) for multiple incidents has imposed tremendous pressures and difficulties for most emergency departments.

- **Challenge 3: Uncertainty**

During the resource deployment process of an emergency incident, the uncertainty of the information of required resources (i.e. resource availability, location, etc.) could cause a significant problem for searching appropriate resources to conduct the rescue operation.

Traditional centralised resource deployment approaches usually assume that it is possible to acquire global information of all available resources to generate an optimal resource deployment plan [Cho+14; GKR11; Zay+13]. However, in metropolitan regions, rescue resources are distributed over an extensive area with different ownership and managed by the different emergency department. In the majority of countries, different emergency departments usually have their own resource management systems with different regulations to control their rescue resources [CPL14]. Implementing a centralised system to manage all resources that come from different emergency departments would be impractical to be realised in real-world environments. Besides, some emergency departments might have sensitive resource information that they do not want to share with other departments. Therefore, it is merely impossible to ac-



quire and manage all resources with global information, which increases the difficulty of generating the optimal resource deployment plans for emergency incidents.

- **Challenge 4: Variability**

During the processes of resource deployment and rescue operations for emergency incidents, one of the significant concern is the dynamic changes caused by incident variation, task variation and execution variation [Tur+04; She10; LL13; Joh15], which could introduce many unknown factors during rescue operations and cause a large impact on resource deployment plans.

More precisely, the incident variation mainly refers to the total number of emergency incidents in an environment could constantly change as new emergency incidents could arrive unpredictably during the resource deployment processes of old emergency incidents.

The task variation mainly refers to the total number of the resource deployment tasks and the detail of task requirements in an emergency incident (response) could change dynamically as time passed. For example, the emergency response of a small house fire may only have one task that requires fire and rescue service at first. But when the fire has spread and people get burned by the fire, a new task that requires medical service needs to be added to the emergency response. Furthermore, a medical emergency could change from a non-life-threatening situation to a life-threatening situation due to the deterioration of a patient's health condition, which requires immediate ambulance transportation with shorter task deadline comparing to the original situation.

The execution variation mainly refers to the execution of a resource deployment plan could change dynamically during an emergency response process. For example, an on-mission ambulance might fail to arrive an incident location before its required deadline due to a traffic jam and emergency operators might need to assign the resource deployment plan to another ambulance that

is capable of arriving the incident scene on time.

## 1.4 Research Issues and Objectives of This Thesis

Based on the four major research challenges of the resource deployment in emergency management, this thesis proposes five primary research issues that need to be addressed to improve the current emergency resource deployment research, which are described as follows.

- **Issue 1: Effectively modelling a resource deployment problem.**

As described in Subsection 1.3, the information involved in an emergency resource deployment problem usually included environment, emergency incidents, resource deployment tasks, rescue resources, resource deployment objectives and constraints. To develop an efficient and effective resource deployment strategy, it is important to integrate all necessary information into a mathematical model. Nevertheless, most of the existing emergency resource deployment models only focus on modelling some particular aspects of the information that involved in the emergency resource deployment problem [LIB08; Lam+15; LL13], which can significantly limit their usages of handling different types of real-life emergency situations. Therefore, how to reasonably and practically represent and model a resource deployment problem by considering this large amount of information and domain knowledge is always a critical step before any resource deployment strategy can be developed.

- **Issue 2: Generating optimal resource deployment plans by using decentralised manner.**

Centralised resource deployment systems usually suffer from the problems of single point failure in emergency management or incapable of managing rescue resources cross heterogeneous resource systems from different emergency departments. Therefore, decentralised resource deployment approaches need to be designed to handle emergency incidents in metropolitan regions. How-

ever, in decentralised systems, agents usually only have the local view of their environments, which can significantly increase the difficulty of generating the optimal resource deployment plan [CH00; Men+14]. Thus, how to properly design agents coordination and cooperation strategies are extremely important for decentralised resource deployment approaches to generate optimal solutions.

- **Issue 3: Deploying resources to multiple emergency incidents simultaneously.**

In metropolitan regions, multiple emergency incidents can happen concurrently and each of these incidents might have different resource requirements and constraints. A resource deployment strategy needs to reasonably deploy limited resources to these incidents according to their deadlines, severities or the total social welfare. Furthermore, in a multi-incident situation, a resource contention problem can be easily occurred, which refers to two or more emergency incidents require same resources at the same time and cause a resource deployment conflict. In order to effectively solve such a resource contention problem, a multi-incident resource deployment strategy needs to know how to locate extra resources or re-scheduling resources.

- **Issue 4: Dynamic resources deployment to emergency incidents.**

Dynamic decision making is extremely necessary for an emergency resource deployment strategy. An emergency incident might require different rescue resources at a different time, and an emergency resource deployment strategy needs to continually monitor the status of the emergency incident and its surrounding environment and generates new resource deployment plan when necessary. Besides, a rescue resource's routing plan might need to be constantly recalculates based on the real-time traffic information, thus the resource deployment time can be minimised. In current research, although many useful approaches have been proposed to handle different dynamic resource deployment problems during an emergency response process [BJL06; LL13; Bil+14;

LIB08; SL06; GKR11], none of them provide comprehensive resource deployment mechanisms by taking the consideration of incident variation, task variation and execution variation, which may lead to ineffective usage of resources and even the failure of rescue operations. Therefore, how to develop appropriate dynamic resource deployment mechanisms to consider different types of variations is an important issue needs to be addressed during an emergency resource deployment process.

- **Issue 5: Designing emergency response system for real-world situations.**

In modern emergency response process, response codes play a significant role in deploying appropriate rescue resource for emergency incidents with different attributes [HLH16; Su+14; SL16]. In real-world emergency situations, emergency departments in different countries around the world might use different response codes to handle their local emergency incidents. For example, in East Anglia, United Kingdom, incidents are classified by grade-based emergency response codes, while in Queensland, Australis, incidents are labelled by number-based emergency response codes [18c]. Therefore, how to design an emergency response system with the flexibility, extendability and practicability that can be used by different emergency response codes in different regions and countries is an essential issue needs to be considered for a real-world deployable system.

Considering the above five research issues, this thesis sets up five research objectives, which are described as follows.

- **Objective 1: To develop an agent-based resource deployment framework.**

The framework should be capable of modelling and representing all important features and relationships among rescue resources, emergency incidents, resource deployment objectives, constraints and environmental information.

- **Objective 2: To develop a resource deployment approach for handling a single emergency incident.**

This approach should be capable of deploying resources to a multi-constraint emergency incident through decentralised manner based on the previous designed agent-based resource deployment framework and generating optimal deployment solution by using advanced optimisation algorithms, such as the constraint programming or the linear programming.

- **Objective 3: To develop a resource deployment approach for handling concurrent emergency incidents.**

This approach should be capable of deploying resources to concurrent emergency incidents simultaneously with the resource usage rationalisation mechanism and effectively addressing the resource contention problems between these incidents to avoid resource deployment conflicts.

- **Objective 4: To integrate dynamic decision-making mechanisms into concurrent resource deployment approaches.**

These mechanisms will be combined with the proposed concurrent rescue resource deployment approach and are capable of handling the dynamic problems caused by incident variation, task variation and execution variation during an emergency resource deployment process.

- **Objective 5: To design a smart emergency response system.**

This system should be practical enough to be deployed in real-world environments with high extendability and adaptability and should be capable of generating resources deployment plans based on different emergency response codes for real-life emergency situations.

## 1.5 Research Contributions of This Thesis

Based on the research objectives introduced in Section 1.4, there are four major contributions of this thesis, which are described as follows.

### 1. An innovation approach for optimal resource deployment in emergency management

An agent-based decentralised resource deployment approach is proposed in this thesis for handling a single emergency incident in metropolitan regions, which contains the solutions for solving the research **Objectives 1** and **2**. Generally, the proposed approach first converts the resource deployment problem of a single emergency incident to different resource deployment tasks. Then, multiple agents simultaneously propose their resource deployment proposals to these tasks. Finally, for each task, the domain transportation theory is used to combine these proposals and generate the optimal solution that minimises the total deployment cost.

This approach includes the following major contributions.

- The proposed approach introduces a comprehensive mathematical model to represent important information that required by most resource deployment systems.
- The proposed approach is designed to effectively deploy distributed resources with different attributes in a decentralised manner.
- The proposed MAS framework is designed to simultaneous deploy different types of resources to multiple interrelated tasks
- The proposed resource deployment algorithm is capable of selecting resources with appropriate deployment time and money expenditures and generating an optimal resource deployment proposal according to an emergency incident's severity.

## **2. An agent-based dynamic resource deployment approach for improving emergency responses to concurrent incidents**

An agent-based resource management approach is proposed in this thesis to provide emergency responses to concurrent emergency incidents simultaneously, which contains the solutions for solving the research **Objectives 3** and **4**. The proposed approach can efficiently coordinate multiple emergency departments with different agents to rationalise the usage of their limited rescue resources for multiple incidents and automatically adjusting resource deployment plans through resource coordination, rescheduling and releasing to adapt the variations that occur during the emergency rescue operations.

This approach includes the following major contributions.

- The proposed approach provides three adaptive resource deployment mechanisms for concurrent emergency incidents to handle the dynamic changes in terms of incident variation, task variation and execution variation, respectively.
- The proposed approach is designed to generate the optimal solutions for concurrent incidents according to resource money expenditure and deployment time, which cannot only help public emergency departments to effectively reduce the average response time of their services, but also rationalise the usage of expensive resources for appropriate incidents.

## **3. A deployable smart emergency response system for real-world emergency situations**

An agent-based resource management system is proposed in this thesis to provide smart emergency response to concurrent emergency incidents simultaneously, by automatically and efficiently coordinating different emergency departments with different agents to generate effective and flexible resource deployment plans based on the attributes of emergency incidents and rescue resources. The proposed multi-agent emergency response system is the solution

for the research **Objective 5**.

This system includes the following major contributions.

- The proposed system is designed to flexibly generate resource deployment solutions for multiple incidents according to different objectives, which helps public emergency departments from different regions or countries to effectively rationalise their resource usages based on their local emergency response codes.
- The architecture of the proposed system is capable of automatically coordinating multiple departments to collaborate with each other and dispatch resources from different emergency facilities by the consideration of a variety of practical implementation factors and difficulties in the real-world environments.

## 1.6 Organisation of This Thesis

**Chapter 1** introduces background knowledge of emergency management, agent-based resource deployment in emergency management, resource challenges of emergency resource deployment, the objectives and contributions of this PhD study.

**Chapter 2** reviews current related work in agent-based resource deployment in emergency management, which include resource deployment modelling in emergency response, resource deployment for concurrent emergency incidents, dynamic resource deployment strategies and real-world emergency resource management systems.

**Chapter 3** presents an agent-based decentralised resource deployment approach for an single emergency incident in metropolitan regions. The proposed approach aims at not only introducing an innovative mathematical model and agent framework to handle the resource deployment process for emergency incidents, but



also providing the optimal resource deployment plan for an incident with domain transportation theory.

**Chapter 4** presents an agent-based dynamic resource deployment approach for improving emergency responses for concurrent incidents. The proposed approach aims at efficiently coordinating multiple emergency departments with different agents to rationalise the usage of their limited rescue resources for multiple incidents and automatically adjusting resource deployment plans through resource coordination, rescheduling and releasing to adapt the variations that occur during the emergency rescue operations.

**Chapter 5** presents an agent-based resource management system to provide smart emergency response to concurrent emergency incidents in real-world environments. The proposed system aims at generating flexible resource deployment plans with a deployable system architecture based on different emergency response codes, objectives and cost attributes that are used by local emergency departments.

**Chapter 6** provides a conclusion of this thesis and indicates some future work.

In addition, an agent-based emergency resource deployment simulator is also developed and implemented to provide a testbed for the comprehensive evaluations for the proposed models, algorithms and mechanisms in this thesis. The detail of the proposed simulator is given in **Appendix A**.

# Chapter 2

## Literature Review

With the rapid increase of emergency incidents around the world, emergency resource deployment have gradually become a significant research area for improving emergency response results.

In recent years, many useful approaches have been proposed to handle different problems that happen during the process of the emergency resource deployment for emergency incidents and natural disasters, such as game theory-based resource deployment [Kie+09; Tsa+10a; Mye13; JAT13], auction-based resource deployment [LIB08; Su+14; SCL05; SL06], dynamical resource deployment mechanisms [BK16; GKR11; Lam+15; Bil+14], agent-based resource deployment [Pon+10; Muk+17; WHM15; BJL06] and so on.

Furthermore, the advancements in computing, networking and telecommunication technologies have made significant impact in emergency management [Mar+10; Ala+14]. Among which, agent and multi-agent technologies are very popular techniques for the implementation of emergency resource management systems. There have been a number of full scale agent-based emergency response systems developed to support resources deployment for different emergency situations, such as FireGrid [Ber+05], RoboCup Rescue [Kit+99], DEFACTO [MST06], DrillSim [Bal+06] and ALADDIN [Ada+08].

This chapter presents a detailed review of state-of-the-art resource deployment approaches and systems in emergency management, which is organised as follows. Section 2.1 reviews several important resource deployment models which are com-

monly used to handle emergency incidents. Section 2.2 reviews various of current approaches of deploying emergency resources for concurrent incidents. Section 2.3 reviews several approaches with dynamic resource deployment mechanisms for handling different variations. Section 2.4 reviews recent advances of real-world emergency resource management systems and simulators. Finally, Section 2.5 summaries this chapter.

## **2.1 Emergency Resource Deployment Modelling**

As described in research Issue 1 in Section 1.4, the foundation of an emergency resource deployment approach is its resource deployment model. To efficiently and effectively deploy appropriate rescue resources to emergency incidents with different attributes, a resource deployment model needs to reasonably represent the large amount of information that involved in the resource deployment process. This section presents a detailed review of emergency resource deployment models based on incident types and optimisation objectives.

### **2.1.1 Modelling Resource Deployment Based on Incident Types**

In emergency management, the information of emergency incidents and their rescue resources are always closely related to each other. Different types of incidents usually require resources with different attributes to handle their emergency response processes. Dispatching incorrect rescue resources to an emergency incident would decrease the efficiency of an emergency response and even results in the failure of a rescue operation completely [OM98; MH03]. Therefore, correctly modelling the relationships between incidents and resources and their attributes is one of the most significant steps to design an effective emergency resource deployment model. The following reviews include various models for handling two types of the most concerned emergency incidents in the literature, which are medical and criminal

emergency incidents.

- **Medical Incidents**

In various types of emergency incidents, a medical incident is the one type that grabs the most attention by current researchers. Billhardt et al. introduced an agent-based coordination mechanism to improve the deployment of ambulances to patients as well as the redeployment of available ambulances [Bil+14]. In their approach, an ambulance requested by an incident was modelled with two attributes, which were its current position and its operational state (i.e. assigned, occupied and idle), while a medical incident was modelled with the number of injured people, their severity levels and corresponding positions. Based on this information, a Bertsekas auction assignment algorithm was used to determine appropriate ambulances deployment with the minimal total response time in the whole process.

López et al. proposed another agent-based approach to coordinate multiple ambulances to medical incidents [LIB08]. In addition to ambulances and incidents' basic positions, López et al.'s approach incorporated an ambulance driver's trust information into their resource deployment model, in which a driver with a high trust value was expected to arrive on time at a medical incident's location and a low trust driver might delay a rescue operation. The advantage of modelling a driver trust information in López et al.'s approach was that the trust information could be used for avoiding possible mistakes in the estimation of ambulance deployment time to an incident's location.

Lam et al. presented an innovative discrete-event simulation model to reduce ambulance response time for Singapore emergency medical services [Wei+14]. Unlike most of the transitional discrete-event simulation models that are not generalisable and rely on specific assumptions, the Lam et al.'s model utilised multiple geographical information systems technologies and an ambulance's historical travel time data to evaluate different resource deployment strategies so as to reduce the emergency response time of an incident.

Kolios et al. developed a decision support tool to manage various types of medical resources (i.e. crews, vehicles, hospital beds, etc.) for medical incidents [Kol+15]. In their model, the information involved in the resource deployment of an emergency response was divided into five spaces, which were Alert/Activation, EMS on-the-way, Field Management, Patient Transportation and First receiver. Besides, all resources and incidents were modelled in a graph to capture the resource' flow among multiple hospitals and victims. However, their approach made the assumptions that the information of resource flow capacities and the knowledge of the initial availability of resources had to be available globally. These assumptions might limit the applications of their approach since it is difficult to acquire requested information in many real-life situations.

Although the above approaches could somehow improve the efficiency of deploying rescue resources for medical incidents from different perspectives, they might be incapable of deploying a resource with specific functionalities to an emergency incident with particular requirements. Solving this problem is essential for reducing the risk of failure of an emergency response. For example, a life-threatening medical incident usually requires ambulances, equipped with an advanced life support system to maintain victims' life conditions [18b]. Incorrectly dispatching ambulances without the advanced life support system to this incident might cause the death of the victims during the transportation process. In order to solve this problem, an agent-based resource deployment model is proposed in **Chapter 3** to properly modelling the different requirements of emergency incidents and the functionalities of rescue resources and correctly match appropriate resources to these incidents.

- **Criminal Incidents**

In recent years, incidents caused by criminal activities and actions of terrorism have also attracted significant attention in the research of the emergency resource deployment, in which game theory is one of the most popular mathematical models used for deploying security resources to handle criminal incidents. Generally, game theory refers to the mathematical study of cooperation and conflicts between rational decision makers [OR94; Mye13]. In the domain of resource deployment for criminal incidents, game theory is usually used to design randomised security resource deployment strategies under the assumption that adversaries might have the knowledge of the deployment strategies [JAT13; Pit+11; Tsa+10b].

James et al. introduced a security resource deployment system called Assistant for Randomised Monitoring over Routes (ARMOR), which was capable of randomised deploying limited police forces in a rational and unpredictable way to patrol and monitor the security of airports [Pit+08]. More precisely, ARMOR could model the airports' security monitoring and patrolling problems as a Bayesian Stackelberg game. In ARMOR, an algorithm known as Decomposed Optimal Bayesian Stackelberg Solver (BOBSS) was used to find out the optimal security resource deployment plan. The plan could provide the schedule of randomised checkpoints monitoring for police forces, which could appropriately weight the benefits and costs as well as uncertainty over different types of adversaries. By using this resource deployment model, the possibility that terrorists comprehend the police forces' patrol and monitor patterns could be significantly reduced, thus terrorist attack could be effectively avoided. However, this model scales poorly when the security policy requires coordination of many different resources. Besides, the process of calculating the optimal deployment plan requires the enumeration of all possible assignments of resources to security checkpoints, which could be extremely time-consuming.

Christopher et al. proposed a randomised security resource deployment strategy for massive security games, which also modelled a security resource deployment problem as a Stackelberg game [Kie+09]. Unlike ARMOR, which restricted to deploy police forces, the deployment strategies proposed by Christopher et al. was designed to handle security problems that required multiple resources. Besides, in Christopher et al.’s approach, a compact representation schema for multiple resources and an algorithm called Efficient Randomised Deployment of Security Resources (ERASER) were used to fully exploit the advantages of this resource’s representation. With this newly designed resource representation schema, ERASER does not need to enumerate all possible resource assignments to calculate the Stackelberg Equilibrium, thus the time and space requirements of computing the optimal resource deployment plan can be significantly reduced.

Jason et al. proposed a graph-based security resource deployment approach to solve city-wide security problems [Tsa+10a]. More precisely, their approach can model an adversary and a defender, who take actions on a graph (i.e. a city map). The adversary starts at one of the source nodes on the graph and chooses a path to travel to certain target nodes. The defender needs to stop the adversary before reaching a target node through deploying security resources on edges of the graph. Unsuccessful attacks always have a payoff for the defender, while successful ones will cause a penalty to the defender. Similar to the approaches in [Pit+08; Kie+09], Jason et al. also modelled the security problem as a Stackelberg game. In their deployment strategy, linear programming was used to calculate the Stackelberg Equilibrium solution, which provided the optimal security resources distribution to capture the adversary. However, Jason et al.’s approach lacks the ability for concurrently handling of attacks caused by multiple adversaries.

Although using game theory to deploy security resources could somehow prevent terrorists and criminals by predicating resource deployment strategies,

these types of approaches neglected the priority or severity level of an emergency incident, due to the randomised resource deployment strategies. This might result in an irrational resource distribution during the process of handling incidents with different severity levels. The proposed resource deployment model in **Chapter 3** is capable of deploying resources with different attributes based on the severity level of an emergency incident to rationalise the usage of limited rescue resources to the incident.

In summary, the resource deployment models reviewed in this section could effectively handle the emergency response problems for their corresponding types of emergency incidents. In general, these models were specially designed for one type of incidents and some of them could only be used to deploy one type of resource, which significantly restrict their usages in the most of real-life emergency situations. In **Chapter 3**, the proposed high-level resource deployment model is designed to efficiently handle different types of emergency incidents that require multiple resources from different emergency departments or services so as to solve research **Issue 1** identified in Section 1.4

### 2.1.2 Modelling Resource Deployment Based on Objective Optimisations

In emergency management, the ultimate goal of resource deployment is to select appropriate rescue resources to fulfil predefined optimisation objectives. Normally, the optimisation objectives of resource deployment for an emergency incident are usually based on the priority or severity level of an incident. For example, an emergency incident with high priority might need to set up the objective to optimise its resource deployment time, while a low priority incident might consider to set up the objective to optimise its money expenditure. This section presents a detailed review of approaches and systems that contain optimisation models for single-objective and multi-objective resource deployment optimisation.



### • Single-Objective Optimisation

In emergency management, *resource deployment time* refers to the time that an incident call is received to the time that rescue units can arrive the incident location. The resource deployment time is one of the most significant objectives to be optimised for saving victims' lives. Lam et al. introduced a model to optimise the response time for ambulance resources, in which geographical information system-based analysis was used to determine ambulances deployment plans based on ambulance-call records and a centralised dispatching unit was used to deploy ambulances to emergency incidents [Lam+15].

Beatriz et al. proposed a multi-agent model for improving the response time of ambulances [LIB08]. Their model consisted of multiple ambulance-team agents and an ambulance-coordinator agent. The ambulance-coordinator agent was responsible for collecting the information of incidents from external agents or human operators and assigns each of these incidents to appropriate ambulances. The optimal assignment problem was solved by using an auction mechanism, which was based on contract-net protocol and a winner determination algorithm. The contract-net protocol is a high-level inverse auction protocol, in which the auctioneer proposes some tasks to bidders to bid. One of the major advantages of using a contract-net protocol is that it allows distributed coordination of agents in a multi-agent system [Smi80; XW01]. In Beatriz et al.'s model, the ambulance-coordinator agent plays the role of the auctioneer by proposing the requirement of an emergency event to the ambulance team agents.

Silvia and Beatriz proposed a reverse combinatorial auction model for tasks and resources deployment in various of emergency rescue scenarios [SCL05]. Silvia and Beatriz's optimisation model was very similar to the resource deployment model proposed by Beatriz et al [LIB08], which also had a coordination agent and multiple rescue team agents. The major differences between these two models are the auction protocols and winner determination algorithms.

Silvia and Beatriz’s model adopted reverse combinatorial auction protocol, in which the rescue team agents gathered local rescue information from incidents’ areas and then passed the rescue tasks to the central coordinator agent. After receiving a certain number of tasks, the central agent returned the complete list of tasks to each rescue agents. With the information of task list, rescue agents sent bids, which consisted of a set of tasks that they were interested in, to the coordinator agent. Finally, the coordinator agent calculated the winner by using the winner determination algorithm and sent the results to corresponding rescue agents. Since this approach only used distance to optimise resource deployment time, it lacked the consideration of a resource’s moving velocity. Further, since rescue agents could only execute rescue tasks sequentially, it might cause the incompleteness of some rescue tasks, placed in the later positions of the task queue.

Later, Silvia et al. further improved the working efficiency of their previous designed resource deployment model [SCL05] in their new work [SL06]. More precisely, in [SL06], the newly designed model had incorporated a bid-tree formulation algorithm, which was capable of generating a binary tree that could support content-based lookup of bids. The bid-tree was used to organise all bids that submitted by rescue team agents. Once the bid-tree was generated, a modified A \* algorithm [Yao+10] was used to search for the winner bids through the tree.

In addition to the *resource deployment time*, the *resource money expenditure* is also an essential objective that needs to be considered to help emergency departments to control their budget. For example, Liu and Liang introduced a discrete time-space network model to deploy medical resources based on the incident demands in an epidemic outbreak [LL13]. Liu and Liang’s model was designed to minimise the total money expenditure of resource distribution crosses the proposed three-level medical logistics network.

Furthermore, Sung and Lee introduced an approach for deploying medical

resources in mass casualty incidents [SL16]. In their approach, the primary optimisation objective was to maximise the number of survivors. Sung and Lee modelled the resource deployment problems for mass casualty incidents as an ambulance routing problem. They applied a column generation approach to generate the optimal deployment plan for all available ambulances based on victims' transportation time.

### • Multi-Objective Optimisation

Due to the complexity of the emergency management and the departments involved in the emergency response process, sometime the resource deployment of an emergency incident might need to consider multiple objectives from different perspectives. Gabdulkhakova et al. proposed an agent-based solution for emergency medical deployment by using a service-oriented architecture [GKR11]. In their system, a master node was required to distribute all service requirements to hospital agents, patient agents and transport agents, and a rational resource deployment algorithm was used to select the medical services with the lowest cost to fill the service requirements based on the money expenditure and time.

Chou et al. also proposed an approach to optimise the resource deployment time and money expenditure to large-scale incidents by using a biological-based genetic algorithm [Cho+14]. In their approach, the traditional genetic algorithm was modified by including elite reserve areas, non-linear value conversion and migration mechanisms to effectively generate the optimal solution.

Widener et al. proposed a hybrid agent-based approach of positioning relief teams for large-scale incidents to assist local residents [WHM15]. In their approach,  $p$ -median optimisation model was used to select the  $p$  best relief team locations based on the given positions of all trapped residents and candidate shelters to minimise the emergency response distance and maximise the task completion number.

Overall, most of the work in both single-objective and multi-objective optimisations can be classified into centralised and decentralised resource deployment modelling. In the majority of centralised approaches or systems, a master node is used to interact with a set of peripheral nodes, and it is assumed that the master node has global information to take appropriate decisions [Tar04]. The advantages of using centralised system control, such as resource deployment optimisation with global information, simplified maintenance and the controlling of all peripheral nodes with one master node. However, we should not neglect the shortcomings of using centralised approaches in emergency management. For example, the approaches proposed by Lam et al. [Cho+14], Widener et al. [WHM15] and [WHM15] were designed to optimise the resource deployment for large-scale emergency events with global information. However, in real-world situations, such large-scale emergency situations usually require different types of resources from different emergency departments or companies. These departments and companies might use their customised systems to manage their own resources, which could significantly increase the difficulty of using a single master node to control all available resources. Another critical problem of using centralised approaches in emergency response is the system failure, especially in medically related emergency services. In [GKR11; Lam+15; SL16], all medical resources such as ambulances, are distributed through a centralised node, which can lead to a single point of failure both in decision making and in communication.

Although decentralised approaches have certain difficulties of achieving the optimal resource deployment results comparing to that of the centralised approaches, they are usually more practical to be implemented than that of the centralised approaches in real-world environments. In decentralised multi-agent systems, auction-based approaches are widely used to organise distributed agents, which is mainly because that an auction mechanism allows agents to make their decisions locally while coordinating globally [Ram+08; KKT10; ED00]. However, one major concern of auction-based approaches is that the competitive relationships between agents

in the above auction-based multi-agent systems might not be suitable for modelling the cooperative nature of emergency management. Another potential problem of the above auction-based approaches is that the resource deployment tasks are bid sequentially, which is not efficient.

In order to overcome the above problems of current optimal resource deployment approaches, a decentralised multi-agent framework is proposed in **Chapter 3** to generate an optimal resource deployment plan for a single emergency incident based on the resource money expenditure and deployment time, which is the solution for addressing research **Issue 2** identified in Section 1.4. In the proposed framework, multiple agents with different roles can act cooperatively to propose solutions for resource deployment tasks and the domain transportation theory is used to optimise the final allocation proposal by considering the solutions of the entire set of participating agents.

## 2.2 Emergency Resource Deployment for Concurrent Incidents

As described in research Issue 3 in Section 1.4, in an open environment, the occurrence of multiple incidents could overlap with each other. These incidents might come with different contents, severity levels and resource deployment requirements and each of them needs to be handled appropriately based on different objectives to improve the emergency response results of all incidents. The rest of this section reviews various emergency resource deployment approaches for handling concurrent incidents.

- **Natural Disasters**

One typical situation that emergency operators need to deploy resources to multiple incidents or locations simultaneously is during post-disaster relief operations, where a large number of casualties are distributed over extended areas and each of them demands relief resources such as food, response personnel, rescue equipment [ZLL12; WHM15; Su+16]. Hu et al. introduced a bi-objective emergency resource deployment approach to dispatch rescue resources to multiple incidents, caused by natural disasters, which used a heuristic particle swarm algorithm to search for the Pareto frontier of resource deployment to these incidents based on efficiency and fairness [HLH16]. Their approach was designed to find a robust resource deployment plans through making the trade-off between the resource deployment efficiency and the fairness under the assumption that the total resource demand was larger than that of the resource reservation. However, their approach could only handle resource deployment with a single rescue resource centre.

Altay proposed a capability-based resource deployment approach for handling concurrent disaster responses [Alt12]. Altay's approach was also designed to deploy resources under the situation that the demand for resource overwhelms the reservation of supplies. In Altay's approach, an integer linear programming model was developed to handle the capability-based multi-resource multi-location deployment problem based on resource deployment time, which used a nationwide resource database to search for resources with appropriate capability for multiple disaster areas.

Su et al. presented another linear-programming-based approach to deploy emergency resources for concurrent incidents in a parallel manner with the consideration of resource deployment time and money expenditure [Su+16]. In their approach, disaster response coalition was first defined as a form of team play. Then, a differential evolution and heuristic-based search algorithm were used to explore disaster response coalitions for concurrent incidents based

on the introduced multiply constrained integer programming model. However, one of the limitations of Su et al.’s approach is that the encoding repair process for their heuristic search algorithm only focused on constrained rescue resources without the consideration of the constraint in response time.

Zhang et al. also tried to address the multiple-resource and multiple-location emergency resource deployment problem by a linear programming method [ZLL12]. Comparing to Altay’s work [Alt12] and Su et al.’s work [Su+16], one of the significant innovation of Zhang et al.’s approach was that their approach estimated the possibility of the occurrence of the secondary disasters that might be triggered by the primary incidents, and used the estimation results to set up a priority of rescue preferences for each location where the second disasters might happen. However, Zhang et al.’s approach assumed that only one secondly disaster could happen at one time without the consideration of the situations where multiple secondly disasters could occur simultaneously.

- **Emergency Incidents**

In addition to the multi-incident situation, caused by natural disasters, unpredictable emergency incidents, such as house fires and vehicle accidents can also occur concurrently at a metropolitan region. These emergency incidents normally require resources come from emergency facilities such as police stations, hospitals, fire stations, etc [Che+08; Che+05]. Hawe et al. proposed an approach to deploy ambulances and fire engines for a hypothetical two-site emergency incidents [Haw+15]. In their approach, an agent-based simulation was used to calculate an efficient resource deployment plan to minimise the total hospital transportation time for critically injured casualties based on the distribution of casualties across the two incident sites. However, Hawe et al.’s approach was designed to handle only two incidents at the same time, which significantly limited its applications for most real-life multi-incident situations.

Gabdulkhakova et al. introduced an agent-based solution for deploying trans-

portation vehicles for victims that involved in mass casualty incidents, in which a mathematical model was developed to rationalise the usage of different vehicles for concurrent incidents based on the victim number of each incident, vehicle transportation time and money expenditure [GKR11]. However, Gabdulkhakova et al.'s approach assumed one vehicle could only transport one victim at one time to simplify the calculation of the vehicle deployment plans. Sung and Lee's approach was also designed to handle the ambulance deployment for mass casualty incidents [SL16]. In their approach, victims who came from different incident sites were assigned with different priorities based on a resource-constrained prioritisation schema, and an optimal resource deployment algorithm was designed to calculate the most efficient victim transportation orders and destination hospitals among concurrent incidents to maximise the number of expected survivals.

In summary, although the above approaches can effectively handle the emergency resource deployment for concurrent incidents based on different strategies or objectives, they fail to address the possible resource contention problems during the process of multi-incident emergency responses [YH95], which could easily occur in a decentralised system. More precisely, in decentralised systems, large amount of distributed components with different goals usually try to interact with each other to solve some common tasks with their local information. In such a system, resource contention or conflict between multiple components is a critical problem to address and can happen commonly, mainly due to these components' incomplete knowledge of global environments or information exchange delay. It is no doubt that resource contention problems in decentralised systems can lead to the increase of system instability and the decline of system performance.

In **Chapter 4**, an agent-based approach to handle concurrent emergency incidents in a decentralised manner is proposed, which includes detailed algorithms for effectively addressing the resource contention problems between different resource deployment tasks from multiple incidents. The proposed approach in **Chapter 4**



provides a solution for solving research **Issue 3**, identified in Section 1.4.

## 2.3 Dynamic Emergency Resource Deployment Approaches

As described in research **Issue 4** in Section 1.4, a resource deployment approach that possesses the capability of dynamically adjusting its plans based on the variations of the open environment is critical for the emergency responses to concurrent incidents. In current literature, many useful approaches have been proposed to handle different resource deployment problems caused by incident variation, task variation and execution in emergency response.

For the incident variation, Buford et al. developed a Belief-Desire-Intention multi-agent system to handle large-scale disaster relief operations with the uncertain incident number [BJL06]. Liu and Liang introduced a discrete time-space network model to deploy medical resources based on the dynamic incident demands in an epidemic outbreak [LL13]. ZayasCabán proposed a medical service deployment approach for the emergency responses of large-scale incidents in a region consisting of multiple cities [Zay+13], in which a deterministic resource deployment model and a statewide decision-maker were used to control rescue vehicles from unaffected city areas to affected city areas dynamically. However, these approaches usually require global information in terms of resources and incidents to handle concurrent emergency responses. This requirement is extremely difficult or impossible to be met in dynamic environments with different levels of variations.

For the task variation, several auction-based approaches [Bil+14; LIB08; SL06] were proposed to dynamically distribute ambulances to appropriate rescue tasks with the consideration of multiple objectives. By using auction mechanisms, these approaches possess certain advantages in emergency resource deployment, such as effective resource management with decentralised manners, optimal resource assignment for tasks with fast response time, etc. However, in auction-based approaches, multiple agents usually compete with each other to win bids, which might not be

the best approach to modelling the cooperative relationships between different emergency departments.

For the execution variation, Gabdulkhakova et al. introduced a centralised multi-agent system to rationalise the resource deployment for mass casualty incidents with the consideration of resource contention problems [GKR11]. Lam et al. applied geographical information system-based analysis to dynamically reassign ambulance deployment locations to balance ambulance availability and demands [Lam+15]. Gong and Batta also introduced an iterative procedure for deployment and redeployment of ambulances to casualty clusters in a post-disaster relief operation based on discrete time policy [GB07]. One of the drawbacks of the discrete time policy is that some clusters might need to wait for another resource deployment iteration before they get serviced if the clusters are reported after the redeployment of ambulances.

In summary, although most of the above approaches can provide promising solutions to address certain dynamic resource deployment problems in emergency management, yet none of them provides comprehensive resource deployment mechanisms by taking the consideration of all the significant problems caused by incident variation, task variation and execution variation in emergency management, which may lead to ineffective usage of resources and even the failure of rescue operations. In **Chapter 4**, three agent-based dynamic resource deployment mechanisms are proposed to efficiently address various dynamic changes in terms of incident variation, task variation and execution variation during the resource deployment process, which provide solutions for solving research **Issue 4**, identified in Section 1.4.

## 2.4 Emergency Management Systems and Simulators

The merging of computing and telecommunication and wireless network technologies have been witnessing during this PhD study, and the merging of these technologies brought significant benefits to the resource deployment for emergency incidents and natural disasters. This section reviews some of the most important emergency man-

agement systems and simulators that have been designed or developed to improve the efficiency and effectiveness of emergency resource deployment for both nature disasters and emergency incidents.

- **Natural Disasters**

During the emergency management for natural disasters, preparedness phrase is extremely critical of reducing the effects caused by the disasters. Generally, the preparedness phrase in emergency management refers to the process of planning and pre-deployment of rescue and response resources for the purpose of reducing vulnerability to disasters, mitigating the impact of disasters and responding more efficiently during disasters. In order to effectively conduct the preparation work for a natural disaster, it is important to measure and predict the possible happening locations, affecting areas, severity and resource demands of the natural disasters [Kap08].

Dawson et al. proposed a simulation system for the resource deployment of risk-based flood incident management [DPW11]. The proposed system was capable of providing quantified modelling of flood incidents and predicating vulnerability of individuals to flooding under different storm surge conditions. More precisely, the prediction of the probability of death or severe injury as a consequence of exposure to flooding was mainly based on the calculation of the depth, the rate of water level increases and the velocity of floodwater. In the proposed simulation system, flood flows were modelled by continuity and momentum equations. Besides, agents were used for the traffic simulation in flood vulnerable areas. Based on the vulnerability calculation and traffic simulation, rescue and response resources were deployed accordingly.

Chang et al. developed a decision-making system for the preparation and resource deployment of flood emergency [CTC07]. In their system, a flood emergency preparation problem with uncertain was formulated as two stochastic programming models. These modules used a number of decision variables to determine the deployment plan of rescue resources for a flood disaster, which

included the structure of rescue organisations, the distributions of rescue resources, locations of rescue resource storehouses and allocations of rescue resources under capacity restrictions. Besides, this system also had the ability to predict the resource demands for a flood disaster. Unlike the flood management system of Dawson et al. [DPW11], which used hydrodynamic simulation to predict the resource demands for a flood disaster, the Chang et al.'s system mainly used the geographic information system and flooding potential maps to predict the possible locations of rescue demand points and the required amount of rescue resources.

Li and Tang developed an emergency-logistics-planning system for several disaster's resource deployment [LT08]. Their system consisted of six subsystems. More precisely, a geology subsystem was used to predict geological subsidiary disasters according to geological spatial information, a weather subsystem used to provide weather prediction and an epidemiology subsystem was used to forecasts disease epidemics after disasters. Besides, a pollution subsystem and a transportation system were used to monitor water pollution and traffic network, respectively. By using the information provided by these predication and monitoring subsystems, medical and rescue subsystem can deploy rescue resources beforehand.

Apart from the prediction-based resource deployment system, agent and multi-agent systems have been widely developed to handle the emergency response problems for different types of natural disasters due to their abilities of collective decision making, decentralised cooperation and collaborative planning in large-scale open environments [BMM11]. In recent years, a number of emergency management systems and simulator have been implemented based on agent and multi-agent technologies, such as RoboCup Rescue [Kit+99], DEFACTO [MST06] and ALADDIN [Ada+08].

In detail, Kitano et al. developed an user-centric simulation system for modelling large-scale search and rescue operations in metropolitan regions, which

was called RoboCup Rescue [Kit+99]. In the RoboCup simulation system, an agent's basic tasks included to collect, store and evaluate information. Multiple agents needed to cooperate with each other to explore disaster space to find victims and predicted the victims' lifetime so as to optimise the agents' rescue actions sequence using genetic algorithms. However, RoboCup system did not support to adjust the agent plans when a situation changes.

Marecki et al. introduced another user-centric simulation system for disaster rescue operations by using the DEFACTO coordination system [MST06], which incorporated 3D visualisation omni-viewer and human-interaction reasoning into a unique high fidelity system. Generally, this system was mainly designed to provide the calibration of human-agent transfer-of-control strategies and real-time emergency response supports and training mechanisms. Since DEFACTO system requires high bandwidth communication channel for agents to exchange information, this might cause the communication delay between agents under a disaster situation.

Adams et al. proposed an intelligent agents system for disaster management called ALADDIN [Ada+08]. Generally, ALADDIN was composed of autonomous, reactive and proactive agents. These agents were grouped into multiple coalitions and assigned with different tasks by using neural-network-based optimisation technologies. The agents collaborated with each other based on multi-dimensional trust and reputation model to achieve a global goal.

Overall, predication-based resource deployment systems are mainly used to pre-deploy rescue resources for large-scale natural disasters in advance to effectively reduce the impact of these disasters [DPW11; CTC07; LT08], which might not suitable for the resource deployment of unpredictable emergency incidents that happen randomly in urban areas. Agent-based disaster management systems and simulators could provide effective coordination mechanisms to simulate multi-department collaboration [Joh15; Cho+14] during

disaster response, but lack the appropriate approaches to rationalise and balance the usage of limited rescue resources based on the requirements of the local governments.

- **Emergency Incidents**

The advances in science and technologies have been changing the traditional process of human-based emergency resource deployment. In modern emergency responses, computer-aided dispatch (CAD) systems play a critical role in rescue resource management. Currently, majority of CAD systems, used by emergency departments around the world, have provided three major functionalities for emergency operators, which are (1) sending dispatch orders to rescue resources; (2) recording the information of emergency incidents and dispatched resources into databases and (3) tracking dispatched rescue units [Saa06].

Although using CAD systems could simplify the management of rescue resources [Saa06], they do not possess the abilities to dispatch rescue resources for emergency incidents automatically and emergency operators still need to manually select appropriate resources and deploy them for emergency incidents. According to our analysis of the open database of San Francisco Fire Department [18e], the time used for manually dispatching resources to an emergency incident usually occupy 35% to 45% of the total resource deployment time, which is a time-consuming process for life-threatening incidents such as house fires or vehicle accidents.

In addition to the CAD systems, the research of auto emergency detection systems also play an essential role in decreasing emergency resource dispatch time. One of the most important emergency detection systems was eCall, which was designed to automatically send accident notifications to emergency centres [18d]. OnStar was another in-vehicle safety system, developed by General Motors [18f]. Similar to eCall, OnStar was designed to detect vehicle collisions

and initiate emergency voice calls automatically. Furthermore, OnStar was capable of analysing the vehicle damage status and generating damage reports for emergency operators. By using the vehicle crash information provided by eCall or OnStar, it was estimated that the average resource deployment time of emergency services can be reduced by 40% in metropolitan regions and more than 50% in rural [Mar+10]. Apart from the automatic crash notification systems used by vehicles, the detection of medical emergency incidents has also benefited significantly from the development of an e-health care system [Pau+09].

Besides, agent-based simulators also provide promising platforms for emergency operators to analyse and study the resource deployment for emergency incidents. Berry et al. developed an integrated emergency response system and a fire safety simulator to provide response processes and high-level plans in the built environment, which was called FireGrid [Ber+05]. The architecture of FireGrid was designed based on task-centric I-X agent architecture, which contains four core technologies: (1) fire and building structure modelling; (2) adaptive routing algorithms for wireless sensors; (3) grid computing for sensor-guided computations and data mining for key incidents and (4) command-and-control using knowledge-based hierarchical task network planning techniques. However, one of the limitations of FireGrid is that the resource deployment plans generated by the system cannot adapt to situation changes.

Balasubramanian et al. presented a simulation framework for emergency response drills called DrillSim [Bal+06]. In DrillSim, each agent had a role and a profile, which could be customised for the emergency responses of different types of emergency incidents or disasters. Furthermore, an agent's behaviour was modelled as a discrete process, where agents alternated between awake and sleep states. During the awake state, the agent acquired the information of its surrounding environments and made decisions based on this information using a recurrent neural network.

In summary, the majority of the current agent-based systems might have different problems for handling the emergency responses for incidents in real-life situations. For example, some agent-based systems only focused on addressing the emergency response problems for a single emergency department [LIB08; Bil+14]. Some agent-based simulators primarily focused on analysing or studying emergency response behaviours without providing concrete solutions about how to improve the efficiency and effectiveness of emergency responses for incidents [Bal+06; Ber+05].

To overcome the limitations of current emergency management and response systems, discussed in this section, in **Chapter 5**, an agent-based smart emergency response system is proposed to coordinate multiple emergency departments to automatically and intelligently generate resources deployment plans to emergency incidents with the consideration of multiple objectives, which is the solution for solving research **Issue 5**, identified in Section 1.4. The proposed system in **Chapter 5** is capable of adjusting its resource usage rationalisation schema based only on emergency response codes used by different emergency departments around the world.

## 2.5 Summary

This chapter gave a detailed review of current literature in emergency resource deployment from different perspectives, including emergency resource deployment models, optimisation objectives, multi-incident resource deployment approaches, dynamic resource deployment mechanisms, emergency management systems and simulators. The advantages and disadvantages of these approaches and models were also analysed at the end of each section. Chapters 3 to 6 of the thesis present solutions generated from this PhD study, for solving the five research issues, identified in Section 1.4.



## Chapter 3

# An Innovative Approach for Optimal Resource Deployment to A Single Emergency Incident

In this chapter, a decentralised resource deployment approach is proposed to handle a multi-task emergency incident, which provides solutions for solving research **Issues 1 and 2**, identified in Section 1.4.

The proposed approach contains a resource deployment model to represent important features and relationships among rescue resources, emergency incidents, resource deployment objectives, constraints and environmental information. Furthermore, the proposed approach provides an agent-based resource deployment framework to effectively select appropriate resources without the global information and concurrently generate the resource deployment plans for multiple tasks by considering the severity level of an emergency incident. In the experiment, the proposed approach is tested along with other related approaches, and the experimental results indicate that the proposed approach can efficiently generate the optimal solution in terms of resource deployment time and money expenditure.

The rest of this chapter is organised as follow. Section 3.1 gives the definitions of the resource deployment model used by the proposed approach and the problem description. Section 3.2 introduces the theoretical foundation of the optimal resource allocation. Section 3.3 describes the agent-based solution and the implementation

of the proposed resource allocation approach. Section 3.4 shows the experimental results and provides analysis. Section 3.5 demonstrates the proposed approach in a case study. Section 3.6 summaries this chapter.

## 3.1 Definitions and Problem Description

This section introduces the important definitions used in the proposed resource deployment model and the description of the fundamental problem that the proposed approach is trying to address.

### 3.1.1 Definitions of Domain Knowledge

**Definition 1** (Environment). An *environment* is represented by a city map, which is defined as an undirected graph,  $G = [\mathbb{V}, \mathbb{E}]$ .

- $\mathbb{V} = \{v_1, v_2, \dots, v_i\}$  is a set of nodes, which represent important locations in a metropolitan region.
- $\mathbb{E} = \{e_1, e_2, \dots, e_j\}$  is a set of edges, which represent the paths between the nodes.  $e_j$  is further defined as a two-tuple,  $e_j = (v_o, v_p)$ , and  $v_o, v_p \in \mathbb{V}$  are the nodes that be connected by  $e_j$ .

**Definition 2** (Resource). A *resource* is defined as a seven-tuple,  $res = (nam, rty, ser, fun, rlo, ava, vel, exp)$ , where

- *nam* represents the name of the resource.
- $rty \in \{facility, mobile\}$  represents the type of resources. *facility* refers to unmovable rescue resources, such as fire stations, hospitals, etc, and *mobile* refers to rescue vehicles and personnel.
- $ser \in \{fire \ \& \ rescue, medical, police\}$  represents the type of emergency services that *res* can provide.
- $fun \in \{hospitalisation, ambulance transport, fire fighting, police support\}$  represents *res* 's functionality.

- $rlo \in \mathbb{V}$  represents  $res$  's current location.
- $vel \in (0, +\infty)$  represents  $res$  's average velocity in kilometre per hour (km/h).
- $exp$  represents  $res$  's money unit.

In the proposed approach, set  $\mathbb{REE}$  indicates all resources in  $G$ . Besides, it is assumed that the money expenditure  $exp$  of  $res$  is known by local emergency departments. Furthermore, the set of resource services and functionalities defined above could be extended in real-world applications.

**Definition 3** (Task). A *task* is defined as a three-tuple,  $tas = (dea, ser, \mathbb{TR})$ , where

- $dea$  represents the deadline of the resources required by  $tas$  to arrive at the incident position.
- $ser$  represents the type of emergency service that  $tas$  requires to make response.
- $\mathbb{TR} = \{tr_1, tr_2, \dots, tr_e\}$  represents a set of required resources for completing  $tas$  and  $tr_e = \{rty, fun\}$ .

In the proposed approach, it is assumed that local emergency departments have the knowledge to estimate  $dea$  based on an incident's severity and content.

**Definition 4** (Emergency Incident). An *emergency incident* is defined as a five-tuple,  $eve = (con, \mathbb{SER}, elo, sev, \mathbb{TAS})$ , where

- $con \in \{fire, rescue, loss\ of\ life, damage\ to\ health, security\ of\ person, security\ of\ property\}$  represents  $eve$  's content.
- $\mathbb{SER} \subseteq \{fire\ \&\ rescue, medical, police\}$  represents a set of emergency services required by  $eve$ .
- $elo \in \mathbb{V}$  represents  $eve$  's location.
- $sev \in [1, 2, 3, 4, 5]$  represents  $eve$  's severity, where 1 indicates the lowest severity and 5 indicates the highest severity.

- $\text{TAS} = \{tas_1, tas_2, \dots, tas_k\}$  represents a sequence of tasks that need to be completed for *eve*.

**Definition 5** (Resource Deployment Plan). A *resource deployment plan* for an incident is defined as a two-tuple,  $rap = (eve, \text{RES})$ , where

- *eve* represents an emergency incident.
- $\text{RES} \subseteq \text{REE}$  represents a set of resources that be proposed for completing tasks.

Besides, a *resource deployment plan* for a single task in the incident is defined as a two-tuple,  $rap_k = (tas_k, \text{RES})$  and  $rap_k.\text{RES} \subseteq rap.\text{RES}$ .

In the proposed approach, the cost of resource deployment is calculated by a cost function, which is used to determine the optimal resource deployment plan (i.e. the plan with the minimum cost). Usually, different emergency incidents might need to use different cost functions, which might involve different cost attributes. Here, a cost function is defined, by the consideration of two significant factors of resource deployment in emergency management, i.e., money expenditure and time [Wau99]. More specifically, there is no doubt that the arrival time of emergency services is crucial for the mitigation of the damage caused by an emergency incident. Apart from the resource deployment time, money expenditure also plays a non-ignorable role in emergency management, since the budget for an emergency department is limited in terms of managing large amount of rescue resources to address unlimited emergency incidents. Besides, the resource money expenditure is also closely related to the service quality that a resource can provide or the scarcity of a type of resources [BK16]. For example, a mobile resource with high moving velocity might require high money expenditure. Therefore, improving the reasonability and effectiveness of money usage in emergency management means that more emergency incidents can be processed and more lives can be saved. Apparently, money expenditure and time should have different significance for different incidents. In the proposed resource deployment approach, the importance of money expenditure and time is

determined by the severity of an incident (i.e. *eve.ser*). For a high severity incident, the proposed approach will consider resource deployment time as a more important attribute compared with money expenditure in the cost function. On the contrary, for an incident with a low severity, resource money expenditure will play a more important role than that of deployment time in the cost function.

In the following, the cost function for deploying a single resource to a single task is first defined.

**Definition 6** (Cost Function). A *cost function* for a single resource deployment is defined by Equation 3.1:

$$COR(eve.tas_k, res) = \begin{cases} E^m(res.exp), & \text{if } res.rty = facility \\ (w^t E^t(\frac{DIS(res.rlo, eve.elo)}{res.vel}) + w^m E^m(res.exp)) \times \\ DLINE(eve.tas_k, res), & \text{if } res.rty = mobile, \end{cases} \quad (3.1)$$

where  $DIS(res.rlo, eve.elo)$  is a function that returns the distance of a passable road between resource location  $res.rlo$  and incident location  $eve.elo$ , which could be implemented by various path searching algorithms, such as A\* algorithms [ZC09].  $E^m(res.exp)$  and  $E^t(\frac{DIS(res.rlo, eve.elo)}{res.vel})$  are two evaluation functions that convert the value of a resource's money expenditure (i.e.  $res.exp$ ) and deployment time (i.e.  $\frac{DIS(res.rlo, eve.elo)}{res.vel}$ ) to a normalised value in-between 0 and 1, respectively. These evaluation functions can be implemented based on various normalisation approaches, such as feature scaling [AH01].  $w^t$  and  $w^m$  represent the weighting (importance) of the resource's money expenditure and deployment time, respectively, which are calculated by following equations:

$$\begin{cases} w^t = \frac{eve.ser}{5} \\ w^m = 1 - \frac{eve.ser}{5}. \end{cases} \quad (3.2)$$

Besides,  $DLINE(eve.tas_k, res)$  is a function, used to determine whether  $res$  can

be deployed within  $eve.task_k$ 's deadline, which is further defined by Equation 3.3:

$$DLINE(eve.task_k, res) = \begin{cases} 1, & \text{if } \frac{DIS(res.rlo, eve.elo)}{res.vel} \leq eve.task_k.dea \\ +\infty, & \text{if } \frac{DIS(res.rlo, eve.elo)}{res.vel} > eve.task_k.dea. \end{cases} \quad (3.3)$$

Based on above terms, the cost function for deploying all required resources to a single task is defined by Equation 3.4:

$$COT(eve.task_k, rap_k) = \sum_{res_v \in rap_k.RES} COR(eve.task_k, res_v). \quad (3.4)$$

Furthermore, the cost function for deploying all required resources to all tasks in a incident is defined by Equation 3.5:

$$COE(eve, rap) = \sum_{eve.task_k \in eve.TAS} COT(eve.task_k, rap_k) \text{ subject to } rap_k.RES \in rap.RES. \quad (3.5)$$

### 3.1.2 Problem Description

For an emergency incident  $eve$ , there could be different resource deployment plans. In the proposed approach, the all possible plans for  $eve$  are represented as a set  $\mathbb{RAP}$ . The main objective of the proposed approach is to search for an optimal resource deployment plan  $rap^* \in \mathbb{RAP}$  for  $eve$ , which is formally defined by Equation 3.6:

$$OBJE = \arg \min_{rap^* \in \mathbb{RAP}} COE(eve, rap^*) \text{ subject to } rap^* \in \mathbb{REE}, \quad (3.6)$$

where  $rap^* \in \mathbb{REE}$  means that the proposed resources in  $rap^*.RES$  must belong to the available resources in the environment  $G$ . The incident objective function indicates that the optimal resource deployment plan  $rap^*$  must have the minimum deployment cost in  $\mathbb{RAP}$ . Besides, the proposed approach assumes there is always enough resources in  $\mathbb{REE}$  to be deployed for  $eve$ .

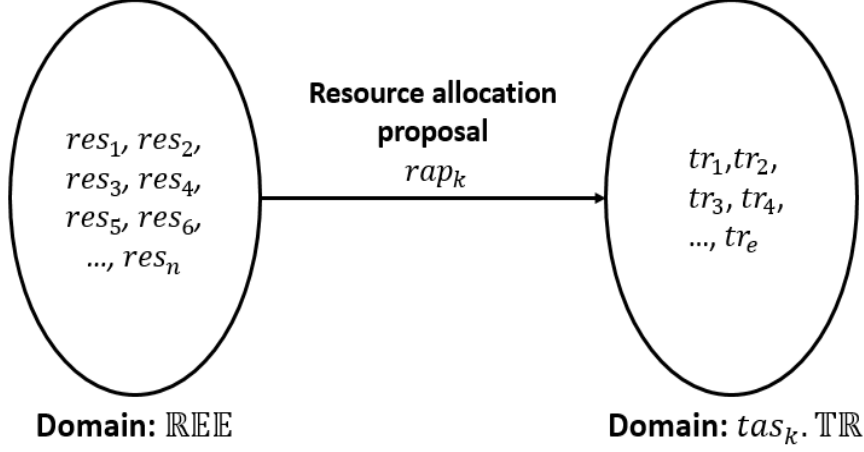
However, due to the fact that an emergency incident usually requires resources with different types and functionalities, searching for a complete  $rap^*$  could be

a complicated process. In order to efficiently solve this searching problem, the proposed approach creates a set of tasks  $eve.TAS$  for  $eve$ . For each task  $tas_k$  in  $eve.TAS$ , it only requires resources that provide the same type of emergency service (i.e.  $res.ser$ ). For example, for a vehicle accident with two tasks,  $tas_1$  only require resources that provide medical service (i.e. ambulances and hospitals), and  $tas_2$  only require resources that provide fire & rescue services (i.e. fire engines and firefighters). Furthermore, each task  $tas_k$  in  $eve.TAS$  has a different or same resource deployment deadline (i.e.  $tas_k.dea$ ). As described in Definition 3, we assumed that the local emergency departments have the knowledge to determine and estimate each task's deadline according to an incident's content and severity. For example, for a house fire incident, a task that requires fire engines should have shorter deployment deadline than a task that requires ambulances, to ensure the fire engines can arrive at the incident scene earlier than that of the ambulances to conduct firefighting. By doing so, the searching of  $rap^*$  for  $eve$  is converted to the searching of  $rap_k^* \in \mathbb{RAP}_k$  for each task  $tas_k$  in  $eve.TAS$ , where  $\mathbb{RAP}_k$  represents all possible or available resource deployment plans for  $tas_k$ . The major design advantage of converting the problem of single incident resource searching into multi-tasks resource searching is to allow different agents to handle different tasks simultaneously, so as to further improve the resource deployment efficiency. Besides, since these tasks have no resource demand conflicts with each other, agents can solve their tasks in a decentralised manner without the need of a centralised control node (see Section 3.3). The objective function  $OBJT$  of resource deployment for a task  $tas_k$  is defined by Equation 3.7:

$$OBJT = \arg \min_{rap_k^* \in \mathbb{RAP}_k} COT(eve.tas_k, rap_k^*) \text{ subject to} \quad (3.7)$$

$$rap_k^*.RES \in \mathbb{REE},$$

### 3.2 Theoretical Foundation of the Domain Transportation for the Optimal Resource Deployment



**Figure 3.1:** Resource Mapping Example

In the proposed approach, domain transportation theory is applied to generate the optimal resource deployment plan for each task in an emergency incident. Generally, domain transportation theory is a linear programming method to generate the optimal solution for multi-objective problems in polynomial time [Liu11], which is advisable to be used in the proposed approach to efficiently find the optimal resource deployment plan in terms resource deployment time and money expenditure. In domain transportation, the resource deployment problem of a task can be described as a resource mapping problem from the available resources in an environment to the required resources of the task (see Figure 3.1), which is formally represented by Equation 3.8:

$$rap_k : \text{REE} \rightarrow \text{tas}_k.\text{TR} \quad (3.8)$$

Apparently, there are many different mapping plans (i.e.  $\text{RAP}_k$ ) from domain  $\text{REE}$  to domain  $\text{tas}_k.\text{TR}$ . In the proposed approach, domain transportation theory is used to find out  $rap_k^*$  for  $\text{tas}_k$ , which can fulfil the objective function  $\text{OBJT}$  (i.e. Equation 3.7).



More precisely, in the proposed resource deployment problem, Equation 3.6 can be described as follows. The resource  $\mathbb{RRE}$  is distributed over the city map  $G$ , which can be regarded as a vector-valued function on the set of nodes  $\mathbb{V}$ . Let  $\mathbb{RRE}_e(y_i)$  denote the amount of resource with functionality  $e$  at location  $y_i \in \mathbb{V}$ . Naturally,  $\sum_{y_i \in \mathbb{V}} \mathbb{RRE}_e(y_i)$  denotes the total amount of type  $k$  resource in  $G$ .

To solve Equation 3.6, it suffices to consider a single task, namely Equation 3.7. For each  $eve.task_k$ , it requires a set of resources  $\mathbb{TR} = \{tr_1, tr_2, \dots, tr_e\}$ , which might have different functionalities, but are provided by a same emergency service. Let  $x_k = eve.task_k$  and  $tr_e(x_k)$  represent the required amount of functionality  $e$  resource at task  $x_k$ . Then  $\sum_{x_k \in \mathbb{TAS}} tr_e(x_k)$  denotes the total required amount of functionality  $e$  resource in such an incident. As we only consider the resource deployment for a single incident, usually the resource is sufficient, which can be formalised as follows.

$$\sum_{x_k \in \mathbb{TAS}} tr_e(x_k) \leq \sum_{y_i \in \mathbb{V}} \mathbb{RRE}_e(y_i). \quad (3.9)$$

Concerning functionality  $e$  resource, for each  $k$  and  $j$ , the cost of transferring the resource at  $y_j$  to the task  $x_k$  is given by Equation 3.1, i.e.  $COR(x_k, y_j)$ . An admissible deployment plan  $rap_k$  for the task of a single incident is a mapping from  $\mathbb{V}$  to  $\mathbb{TAS}$  satisfying the balance condition that for any subset  $E \subset \mathbb{TAS}$ , which can be defined by the following equation:

$$\sum_{x_k \in E} tr_e(x_k) = \sum_{y_j \in rap_k^{-1}(E)} \mathbb{RRE}_e(y_j), \quad (3.10)$$

where  $rap_k^{-1}$  is the inverse mapping of  $rap_k$ . This balance condition means during the deployment process of the optimal plan, the resources in the plan are neither increase nor decrease. The cost of implementing the plan  $rap_k$ , given by Equation 3.4, is equivalent to

$$COT(rap_k) = \sum_{x_i \in \mathbb{TAS}} COR(x_i, rap_k^{-1}(x_i)). \quad (3.11)$$

The purpose of the objective function of an task, i.e. Equation 3.7 is to find an

optimal plan  $rap_k^* \in \mathbb{RAP}_k$  such that

$$COT(x_k, rap_k^*) = \min_{rap_k \in \mathbb{RAP}_k} COT(x_k, rap_k). \quad (3.12)$$

From optimal transport theory, Equation 3.12 can be transferred to the following linear programming:

$$\begin{aligned} \max \{ & \sum_{x_k \in \mathbb{TAS}, y_j \in \mathbb{V}} u(x_k) tr_e(x_k) + v(y_j) \mathbb{REE}_e(y_j) : u(x_k) + v(y_j) \\ & \leq COR(x_k, y_j) \}. \end{aligned} \quad (3.13)$$

Moreover, there exists a maximiser  $(u^*(x_k), v^*(y_i))$  (unique up to a constant) achieving the above maximum, which represents the optimal deployment for task  $x_k$  by using the resource at location  $y_i$ .  $u^*(x_k)$  is called a potential, and  $v^*(y_i)$  is its dual potential. The pair  $(u^*(x_k), v^*(y_i))$  also satisfies a generalised Legendre duality associated with the cost function for single resource  $COR$  (refer to Equation 3.1). Hence, for each  $x_k$ , there exists an unique  $y_i$  such that the equality holds in the constraint, namely  $u^*(x_k) + v^*(y_i) = COR(x_k, y_i)$ , which means the task  $x_k$  requires the resource from location  $y_i$ . Thus, we can construct a mapping  $inv_k^* : x_k \in \mathbb{TAS} \rightarrow y_j \in \mathbb{V}$ . From optimal transport theory [Liu11], by differentiating the above equation we can see that

$$\begin{aligned} \nabla u^*(x_k) &= \nabla_x COR(x_k, inv_k^*(x_k)), \\ y_i &= inv_k^*(x_k) = \nabla_x^{-1} COR(x_k, \nabla u^*(x_k)). \end{aligned} \quad (3.14)$$

In fact,  $inv_k^*$  is exactly the inverse of the optimal deployment plan  $rap_k^*$ . Therefore, to construct  $rap_k^*$ , it suffices to follow the following steps:

1. From the given data  $\{tr_e, \mathbb{TAS}, \mathbb{REE}_k, \mathbb{V}\}$ , formulate the linear programming Equation 3.13;
2. Solve Equation 3.13 to find out a potential  $u^*$ ;
3. Use Equation 3.14 to construct the mapping  $inv_k^* : x_k \in \mathbb{TAS} \rightarrow \mathbb{V}$ , which

implies that task  $x_k$  requires the resource from the area  $REQ := inv_k^*(x_k) \subset \mathbb{V}$ ;  
and

4. Take the inverse, we obtain the optimal deployment plan  $rap_k^* : REQ \rightarrow x_k$ ,  
which can inform the agent how to distribute the resource in the optimal way.

As a remark, when the cost function  $COR$  satisfies the condition that  $\nabla_x COR(x_k, \cdot)$  is injective at each  $x_k$ , from the uniqueness of potential  $u^*$  (up to a constant) and relation (12), the optimal deployment plan  $rap_k^*$  is also unique.

### 3.3 Agent-based Decentralised Resource Deployment

The proposed resource deployment approach for a single multi-task emergency incident is implemented based on agent and multi-agent technologies. This section gives the detailed description of agents' definitions, implementation, resource deployment framework and process.

#### 3.3.1 Definitions of Agents and Implementation

There are four types of agents in the proposed MAS, which are **response agents**, **mobile agents**, **facility agents** and **deployment agents**. The four types of agents are defined as follows.

**Definition 7** (Response Agent). A *response agent* is represented by  $ra$ , which has the information of a specific emergency incident. A response agent has four major functionalities, including:

1. Identifying incident content  $eve.con$  for a new incident  $eve$ ;
2. Identifying the emergency services  $eve.SER$  that is required by  $eve$  based on  $eve.con$ ;
3. Identifying a set of tasks  $eve.TAS$  for  $eve$  based on  $eve.ser$ ;

4. Sending  $eve.TAS$  to a deployment agent.

**Definition 8** (Deployment Agent). A *deployment agent* is represented by  $da$ , which has the information of a specific emergency incident. A deployment agent has three major functionalities, including:

1. Informing an incident's tasks information (i.e.  $eve.TAS$ ) to facility agents that are located in the deployment agent's circle communication area, represented by  $da.com$ ;
2. Combining and generating the optimal resource deployment plan  $rap_k^*$  for a task based on a set of plans (i.e.  $RAP_k$ ) submitted by facility agents;
3. Informing relevant facility agents to execute  $rap_k^*$ .

**Definition 9** (Facility Agent). A *facility agent* is represented by  $fa$ , which has the information of a specific facility resource  $fa.res$  and a set of mobile agents. More precisely,  $fa.MA = \{ma_1, ma_2, \dots, ma_j\}$  represents a set of mobile agents that belong to  $fa$  and  $REF = \{fa.res\} \cup \{ma_1.res, ma_2.res, \dots, ma_j.res | ma_j \in fa.MA\}$  represents all resources under  $fa$ 's management. A facility agent has three major functionalities, including:

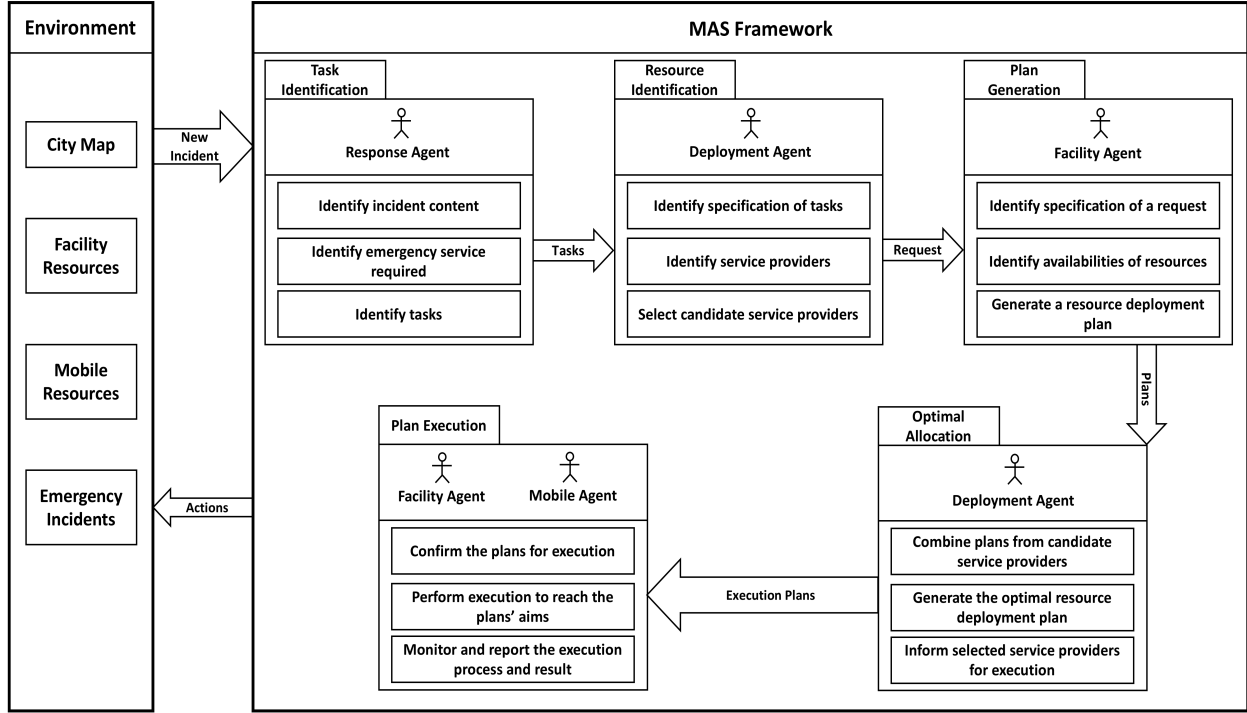
1. Managing a facility resource  $fa.res$ ;
2. Generating resource deployment plans for tasks based on  $fa.REF$ ;
3. Informing the mobile agents which under its control to execute resources deployment commands after receiving the confirmation from a deployment agent.

**Definition 10** (Mobile Agent). A *mobile agent* is represented by  $ma$ , which has the information of a specific mobile resource  $ma.res$ . A mobile agent has two major functionalities, including:

1. Managing a mobile resource  $ma.res$ ;
2. Implementing resource deployment plans after receiving resources deployment commands.

### 3.3.2 Resource Deployment Framework and Process

The proposed resource deployment approach is implemented by a MAS, which includes a task identification module, a resource identification module, a plan generation module, an optimal deployment module and a plan execution module. The framework of the MAS is depicted in Figure 3.2.



**Figure 3.2:** The Framework of the Proposed Resource Deployment System

More precisely, in the task identification module, a single response agent is used to identify resource deployment tasks for an emergency incident according to the emergency service required by this incident. Then, the task identification module passes the tasks to the resource identification module, in which a deployment agent generates multiple task processing threads. Each of the task processing thread is used to identify the resource requirements specification in a task and pass this task to relevant candidate service providers (i.e. facility agents). After that, in the plan generation module, each facility agent that has received a resource deployment task will use domain transportation theory to generate a resource deployment plan based on all resources under its management. All the generated resource deployment plans will be passed to the optimal deployment module, in which the deployment agent will

combine these plans and use domain transportation theory to generate an optimal resource deployment plan for a task. Finally, the optimal deployment plan is passed to the plan execution module, in which relevant facility agents will confirm the detail of the resource deployment plan and inform relevant mobile agents to execute the plan accordingly.

In the proposed multi-agent framework, a facility agent does not responsible for its mobile agents' routing problems or perform the execution plans for its mobile agents. A facility agent only select available mobile resource from its facility and pass execution plans to its mobile resources. The main reason that the proposed framework separate the routing functionalities from facility agents to mobile agents is that it is a mobile agent's responsibility to execute a resource deployment plan in real-world environments. A mobile agent might need to frequently and dynamically reschedule its route to an incident location based on live traffic. If adding the routing functionality to the facility agent, it will significantly increase the computational workload for the facility agent, since a facility agent usually needs to simultaneously manage multiple mobile agents for different incidents. Besides, the communication between a facility agent and a mobile agent will also delay the response time for an incident.

The deployment process is formally described by Algorithm 1.

**Algorithm 1** : Resource Deployment Process

---

```

1: assign  $ra$  to  $eve$ 
2:  $ra$  identifies  $eve.con$ 
3:  $ra$  identifies  $eve.SER$  based on  $eve.con$ 
4:  $ra$  identifies  $eve.TAS$  based on  $eve.SER$ 
5:  $ra$  sends  $eve.TAS$  to  $da$ 
6:  $da$  calculates circle communication area  $da.com$ 
7:  $da$  locates  $\mathbb{FA}$  in  $da.com$ 
8: for all  $tas_k \in eve.TAS$  do
9:    $da$  creates  $\mathbb{RAP}_k$  and  $\mathbb{FA}_k$ 
10:  for all  $fa_i \in \mathbb{FA}$  do
11:    if  $fa_i.res.ser = tas_k.ser$  then
12:       $da$  updates  $\mathbb{FA}_k = \{fa_i\} \cup \mathbb{FA}_k$ 
13:       $da$  sends  $tas_k$  to  $fa_i$ 
14:       $fa_i$  finds  $rap_k^i : fa_i.REF \rightarrow tas_k.TR$ 
15:      if  $tcu \leq PDLINe(tas_k.dea, eve.sev)$  then
16:         $fa_i$  submits  $rap_k^i$  to  $da$ 
17:         $da$  updates  $\mathbb{RED}_k = \{rap_k^i\} \cup \mathbb{RAP}_k$ 
18:      end if
19:    end if
20:  end for
21: end for
22: while  $|eve.TAS| > 0$  do
23:   for all  $tas_k \in eve.TAS$  do
24:    if  $tcu \geq PDLINe(tas_k.dea, eve.sev) \vee \forall fa_g \in \mathbb{FA}_k : fa_g$  submit  $rap_k^g$ 
    then
25:      if  $\mathbb{RAP}_k$  does not contains enough resources for  $tas_k$  then
26:         $da$  expand  $da.com = da.com \times 2$ 
27:        the process goes back to Line 7
28:      else
29:         $da$  finds  $rap_k^* : \mathbb{RED}_k \rightarrow tas_k.TR$ 
30:         $da$  updates  $eve.TAS = eve.TAS \setminus \{tas_k\}$ 
31:         $da$  informs agents to execute  $rap_k^*$ 
32:      end if
33:    end if
34:  end for
35: end while

```

---

The resource deployment process is shown in Algorithm 1 includes six steps, which

are explained as follows.

**Step 1: (Lines 1-5)** When an emergency incident  $eve$  happens, a new response agent  $ra$  is assigned to  $eve$  to identify the emergency content  $eve.con$ . Then,  $ra$  needs to identify the emergency services  $eve.SER$  required by  $eve$  according to  $eve.con$ . For example, when  $eve.con = fire$ , the required emergency services could be  $eve.SER = \{fire \ \& \ rescue, medical, police\}$ . After the emergency service identification,  $ra$  needs to acquire the incident severity (i.e.  $eve.ser$ ) and the resources required by each emergency service, which can be provided by human operators or other external agents. Then,  $ra$  converts each of emergency service to a task. Finally,  $ra$  sends  $eve.TAS$  to a deployment agent  $da$ .

**Step 2: (Lines 6-13)** After receiving  $eve.TAS$ ,  $da$  first needs to calculate a communication area  $da.com$ , which is a circle centred at the incident's location  $eve.elo$  and measured by square kilometres.  $da.com$  is calculated by Equation 3.15:

$$da.com = \pi \times (avv \times (\frac{\sum_{task \in eve.TAS} task.dea}{|eve.TAS|} - tcu))^2, \quad (3.15)$$

where  $avv$  represents the average moving velocity (km/h) of all required mobile resources in  $eve.TAS$  and  $tcu$  represents the current time. In the proposed approach, it is assumed that local emergency departments have the knowledge of the average velocities of resources.

After the calculation of  $da.com$ ,  $da$  needs to locate all facility agents inside  $da.com$ , represented by set  $\mathbb{FA} = \{fa_1, fa_2, \dots, fa_i\}$  (Line 9). Then,  $da$  sends each  $task$  to relevant facility agents in  $\mathbb{FA}$  based on  $task$ 's emergency service requirement  $task.ser$ . At the same time,  $da$  also creates a contact list for the facility agent  $\mathbb{FA}_k = \{fa_1, fa_2, \dots, fa_g\}$  ( $\mathbb{FA}_k \subseteq \mathbb{FA}$ ) and resource deployment plan list  $\mathbb{RAP}_k = \{rap_k^1, rap_k^2, \dots, rap_k^g\}$  for each  $task$  in  $eve.TAS$ .

**Step 3: (Lines 14-17)** After a facility agent  $fa_i \in \mathbb{FA}$  receives  $task$ ,  $fa_i$  uses the domain transportation theory (see Section 3.2) to calculate an optimal resource



deployment plan  $rap_k^i$  based on all available resources that are under  $fa_i$ 's management (Line 14). After the calculation of  $rap_k^i$ ,  $fa_i$  submits  $rap_k^i$  to  $da$  if current time  $tcu$  has not exceeded the task plan deadline. The task plan deadline is calculated by function  $PDLINE(tas_k.dea, eve.sev)$ , which can be defined by local emergency departments based on the detail of  $tas_k$  and  $eve$ . After the submission of  $rap_k^i$ ,  $da$  adds  $rap_k^i$  to  $\mathbb{RAP}_k$ .

**Step 4: (Lines 23-25)** After  $da$  receives  $tas_k$ 's resource deployment plans from all facility agents in  $\mathbb{FA}_k$  or  $tas_k$ 's plan deadline has been reached,  $da$  checks whether  $\mathbb{RAP}_k$  has enough resources for  $da$  to generate a final resource deployment plan to complete  $tas_k$ . If the resources are enough to fulfil  $tas_k$ 's requirements, the process goes to Step 6. Otherwise, the process goes to Step 5.

**Step 5: (Lines 26-27)** If  $\mathbb{RAP}_k$  does not have enough resources to complete  $tas_k$ ,  $da$  expands its original communication area  $da.com$  by doubling its size to be able to contact more facility agents, and then the process goes back to Step 2.

**Step 6: (Lines 29-31)** If  $\mathbb{RAP}_k$  has enough resources to complete  $tas_k$ ,  $da$  uses domain transportation theory (see Section 3.2) to generate an optimal resource deployment plan  $rap_k^*$  for  $tas_k$  based on all resources in  $\mathbb{RAP}_k$ , which is represented by  $\mathbb{RED}_k$  (Line 24). Finally,  $da$  informs relevant facility agents to execute  $rap_k^*$  and remove  $tas_k$  from  $eve.TAS$ . If there are more tasks in  $eve.TAS$ , the process repeats Step 4, otherwise the process ends.

### 3.4 Experiments

In this section, an experiment is conducted to evaluate the performance of the proposed single incident resource deployment approach and the experimental results are presented and the performance of the proposed resource deployment approach is analysed. The experiment focuses primarily on the test of the resource deployment time, money expenditure and cost of an incident when employing the proposed optimal resource deployment approach. The experimental results were compared

with other related emergency resource deployment approaches. The rest of section is divided into two subsections. Subsection 3.4.1 describes the experimental settings and Subsection 3.4.2 demonstrates the experimental results and gives performance analysis.

### 3.4.1 Experimental Setting

In order to effectively analyse the advantages and disadvantages of the proposed approach, the proposed approach is tested along with the other four related agent-based emergency resources deployment approaches for a single incident deployment [Cho+14; GKR11; LIB08; Pon+10]. Briefly, Chou et al. 's [Cho+14] approach was designed to deploy rescue resources to a large-scale emergency situations by using a genetic algorithm; Gabdulkhakova et al. 's [GKR11] approach was designed to generate solutions for emergency medical deployment by using the service-oriented architecture; López et al. 's [LIB08] approach was designed to deploy ambulances for emergency medical incidents by using a contract-net protocol and Ponda et al. 's [Pon+10] approach was designed to deploy tasks to heterogeneous agents by using a distributed auction mechanism. Generally, these four approaches can be classified into centralised approaches [Cho+14; GKR11] and decentralised approaches [LIB08; Pon+10]. In centralised approaches, a centralised agent can not only access the information of all distributed resources but also directly control other agents on the map. Contrarily, in decentralised approaches, because resource information is limited to individual agents, an agent needs to communicate with other agents to exchange information during a resource deployment process. The summary of the four approaches is listed in Table 3.1.

To evaluate our approach in real-world environments, GoogleMap is employed in the resource deployment tests. More precisely, an agent-based emergency resource deployment simulator was designed and implemented to conduct the experiment (see Figure 3.3 in the next page).



**Table 3.1:** Parameters for Test Scenario's Setting

	Approach	Resource Control	Deployment Method	Global Information	Optimisation Objective	Task Processing
1	Gabdulkhakov et al. [GKR11]	Centralised	Rational Deployment	Require	Money & Distance	Sequential
2	Chou et al. [Cho+14]	Centralised	Genetic Algorithm	Require	Money & Time	Sequential
3	López et al. [LIB08]	Decentralised	Contract-net Protocol	Not Require	Time	Sequential
4	Ponda et al. [Pon+10]	Decentralised	Distributed Auction Protocol	Not Require	Money & Time	Sequential
5	Our Approach	Decentralised	Domain Transportation	Not Require	Money & Time	Concurrent

The GoogleMap technology allows the proposed resource deployment simulator to search for the real world information of local facility resources in a particular city, such as the location of fire stations, hospitals and police stations. Since GoogleMap does not provide the information of mobile resources of a facility resource, the experimental parameters of mobile resources were randomly generated based on Table 3.2. The experimental parameters of emergency incidents and tasks were randomly generated based on the range values in Table 3.3 and Table 3.4, respectively.

**Table 3.2:** Parameters for Resource's Setting

<i>rty</i>	<i>vel</i>	<i>exp</i>
<i>mobile</i>	[20km/h – 100km/h]	[50 – 100]
<i>facility</i>	0	[50 – 100]

**Table 3.3:** Parameters for Incident's Setting

<i>con</i>	SER	<i>sev</i>	TAS
{ <i>fire, vehicle accident</i> }	{ <i>fire &amp; rescue, medical, police</i> }	2	[1 – 3]

**Table 3.4:** Parameters for Task's Setting

No.	<i>ser</i>	<i>dea</i>	$ \text{TR} $	<i>tr.rty</i>
[1 – 3]	{ <i>fire &amp; rescue</i> , <i>medical, police</i> }	[5min – 15min]	15	{ <i>mobile</i> , <i>facility</i> }

The resource deployment experiment for each approach was repeated for 1000 times and the average resource deployment cost, time, money expenditure and solution calculation speed of an incident were recorded. The resource deployment cost was calculated by Equation 3.1 and the results of the average resource deployment cost, time and money expenditure were normalised between 0 and 1 by their minimal and maximal values. A high value of the normalised result indicates high cost, while a low normalised value indicates low cost. In each time of the test, the parameters of the emergency incident and its tasks were regenerated, while the parameters of the resources remained the same.

### 3.4.2 Experimental Results and Analysis

The experimental results are shown in Figure 3.4 in the next page, which are resource deployment time test shown in Figure 3.4(a), resource money expenditure test shown in Figure 3.4(b), resource deployment cost shown in Figure 3.4(c) and the calculation time of the optimal solution shown in Figure 3.4(d).

As we can see from Figure 3.4(a), in the three decentralised approaches (i.e. López et al., Ponda et al. and our approach), the approach of López et al. required the shortest resource deployment time (0.26) for an emergency incident in 1000 tests. Our approach performed slightly better than Ponda et al.'s approach (0.36 versus 0.41). Figure 3.4(b) showed that López et al.'s approach costed significant more resource money expenditure than our and Ponda et al.'s approaches. This is mainly because in López et al.'s approach, resource deployment time was used as the only criterion to determine the optimal deployment of resources, but our and Ponda et al.'s approaches took both resource deployment time and resource money expenditure into consideration. Due to this reason, López et al.'s approach eventually

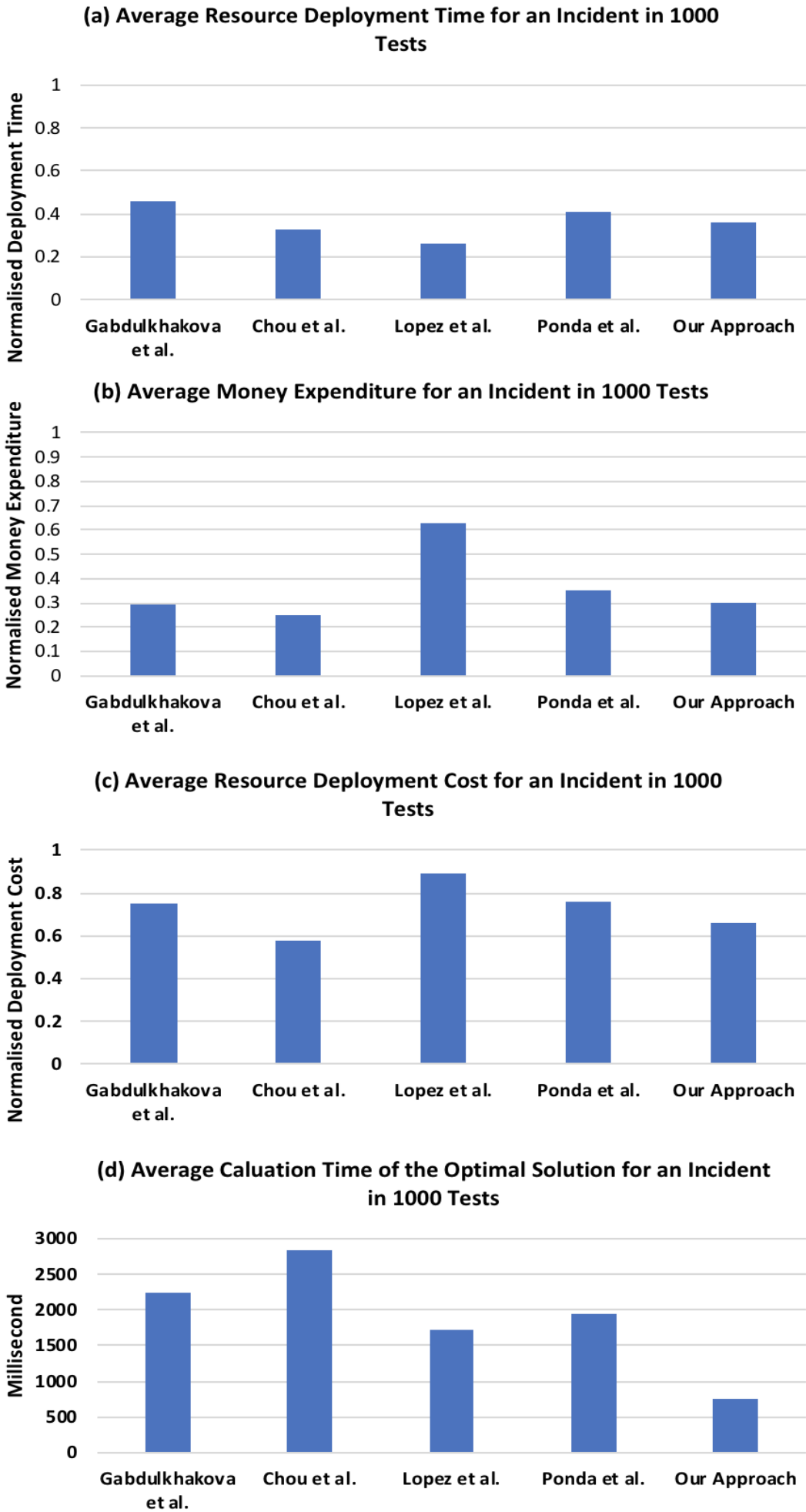


Figure 3.4: Experimental Results

required the most resource deployment cost in the three decentralised approaches. Furthermore, in Figure 3.4(c), we can find that the resource deployment cost of our approach is less than that of the approach of Ponda et al. (0.66 versus 0.76). The reason for generating such experimental results is that Ponda et al.'s resource deployment approach is based on auction mechanism and the relationships between contract agents are competitive. Therefore, the optimal resource deployment plan in Ponda et al.'s approach is the single solution provided by the winner agent of the auction. However, for most emergency situations, it is more reasonable for agents to act cooperatively rather than competitively. In our approach, a deployment agent uses the domain transportation theory to generate the optimal solution based on the resource deployment plans from multiple facility agents, so the optimal solution is a combined solution by integrating the advantages of each facility agent's plan.

Now let us analyse the performance of the centralised approaches (i.e. Gabdulkhakova et al. and Chou et al.). From Figure 3.4(c), we can see that the approach of Chou et al. spent less resource deployment cost than Gabdulkhakova et al.'s approach (0.58 versus 0.75). The major reason of such results is that the optimisation objectives of Gabdulkhakova et al.'s approach are resource distance and money expenditure, which result in some disadvantages of the optimisation of resource deployment time compared with the approach of Chou et al. Comparing these two centralised approaches with our approach, Chou et al.'s approach performed slightly better than our approach in both resource deployment time and money expenditure. This is mainly because a central node was used in their system to acquire all available resources information, but our system was implemented in a decentralised manner.

Finally, from Figure 3.4(d), we can see that our approach outperformed the other four approaches in term of the calculation time of the optimal solution (756 milliseconds), which is not only because that the domain transportation theory was employed in our optimal deployment algorithm, but also because the resource deployment tasks were processed concurrently in our MAS. In contrast, Chou et al.'s

approach required the longest time to generate the optimal solution for an incident (2832 milliseconds), due to the high computation cost of calculating the fitness value function for comparing the advantages and disadvantages of different chromosomes.

In summary, Chou et al.'s approach can efficiently generate the optimal deployment plans by using the centralised resource control system. However, it is almost impossible to access every available resource that come from different emergency departments or companies to control them by using one central node in most real-life situations. In contrast, our approach was designed to deploy resource in a decentralised manner. With the optimal resource mapping plan generated by the domain transportation theory, the resource deployment cost in our approach for an emergency incident is very close to Chou et al.'s centralised approach.

### 3.5 Case Study

In the previous section, we demonstrated that the proposed resource deployment approach could effectively generate a resource deployment plan by considering both resource money expenditure and deployment time. In order to better understand how agents perform their actions during the process of resource deployment and how the proposed approach adjusts its resource deployment strategies in terms of emergency incidents with different severities, a case study is presented in this section. This case study has three major purposes, which are: **(1)** demonstrating agents' activities and information flows during the resource deployment process; **(2)** observing the changes of resources selection of the proposed approach when the severity of an incident is increased from level 1 to level 5; and **(3)** analysing how the proposed approach balances the importance of resource money expenditure and resource deployment time for an incident with different severities.

#### 3.5.1 Case Study Setting

The case study was also conducted on the Google Map. In detail, the emergency incident used for the case study is a fire incident. The parameters of the emergency



incident are listed in Table 3.5.

**Table 3.5:** Parameters for Incident's Setting

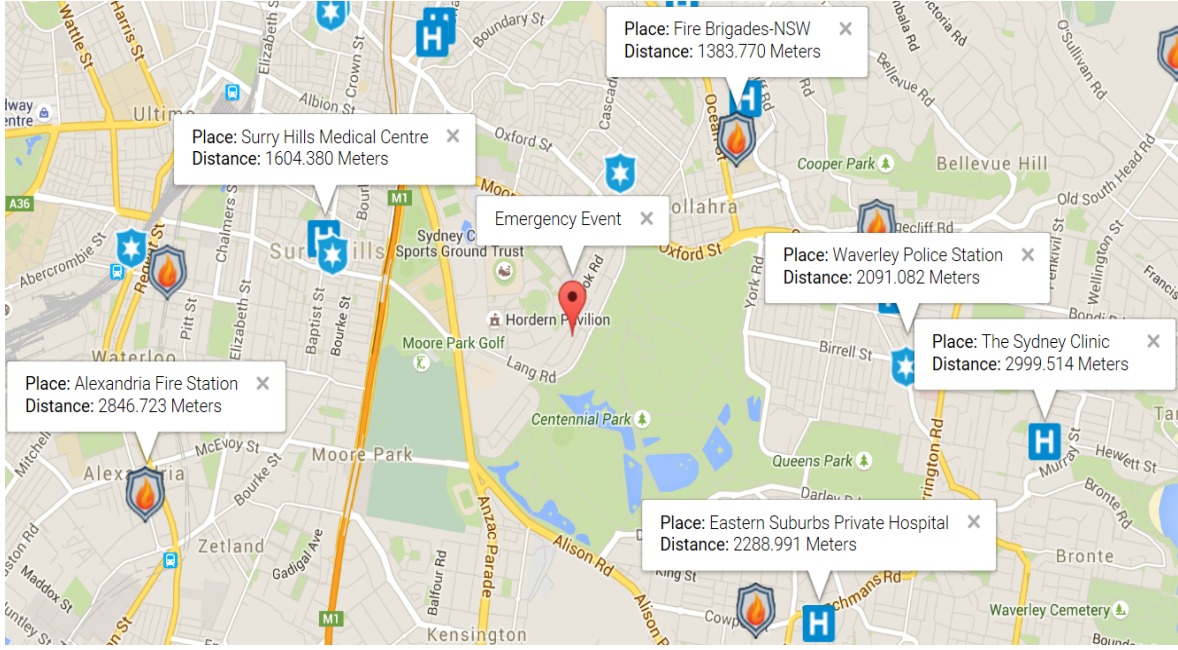
<i>con</i>	SER	<i>sev</i>	TAS
<i>fire</i>	{ <i>fire &amp; rescue, medical, police</i> }	[1-5]	4

There are four resource deployment tasks in this incident, in which Task 1 requires three fire engines, Task 2 requires two police cars, Task 3 requires two ambulances and Task 4 requires one hospital. The detail of tasks' parameters are listed in Table 3.6.

**Table 3.6:** Parameters for Task's Setting

No.	<i>ser</i>	<i>dea</i>	TR	<i>tr.rty</i>	<i>tr.fun</i>
1	<i>fire &amp; rescue</i>	5 min	3	<i>mobile</i>	<i>Fire Fighting</i>
2	<i>police</i>	5 min	2	<i>mobile</i>	<i>Police Support</i>
3	<i>medical</i>	5 min	2	<i>mobile</i>	<i>Ambulance Transport</i>
4	<i>medical</i>	5 min	1	<i>facility</i>	<i>Hospitalisation</i>

The facility resources used in the case study are selected from Google Map, which are two fire stations, two police stations and two hospitals (see Figure 3.5).



**Figure 3.5:** Available Facility Resources

Each of the facility resources has three mobile resources under its management, with different moving velocity and money expenditure (see Table 3.7 in the next page for detail). In the case study, the proposed approach was used to deploy resources to the same emergency incident for five times with different incident severities (from level 1 to level 5). During each time of deployment, resources that had been selected by the proposed approach as the final resource deployment plan were recorded. Besides, the normalised values (between 0 and 1) of the actual resources' money expenditure and deployment time (i.e. the values that have not been scaled by the weighting of money expenditure  $w^m$  and time  $w^t$ ) were also recorded. Furthermore, an activity diagram by using Unified Modelling Language (UML) was provided to demonstrate agent's activities and information outputs during each step of the resource deployment process [RJB04]. Finally, the Pareto optimality test was conducted to verify whether the resource deployment solutions generated from our approach belonged to Pareto frontier [ŠHJ13], which is a widely used method for multi-objective optimisation problems.

**Table 3.7:** Parameters for Resource's Setting

<i>nam</i>	<i>rty</i>	<i>fun</i>	<i>own</i>	<i>exp</i>	<i>vel</i>	<i>DIS</i>	<i>agent</i>
Fire Brigades-NSW (FB)	<i>facility</i>	<i>fire station</i>	none	0	0	1.383 km	f1
ft11	<i>mobile</i>	<i>Fire Fighting</i>	FB	80	90km/h	1.383 km	m11
ft12	<i>mobile</i>	<i>Fire Fighting</i>	FB	70	85km/h	1.383 km	m12
ft13	<i>mobile</i>	<i>Fire Fighting</i>	FB	60	75km/h	1.383 km	m13
Alexandria Fire Station (AF)	<i>facility</i>	<i>fire station</i>	none	0	0	2.846 km	f2
ft21	<i>mobile</i>	<i>Fire Fighting</i>	AF	40	70km/h	2.846 km	m21
ft22	<i>mobile</i>	<i>Fire Fighting</i>	AF	35	65km/h	2.846 km	m22
ft23	<i>mobile</i>	<i>Fire Fighting</i>	AF	30	55km/h	2.846 km	m23
Sydney Mounted Police Stables (SM)	<i>facility</i>	<i>police station</i>	none	0	0	1.512 km	f3
pc31	<i>mobile</i>	<i>Police Support</i>	SM	50	75km/h	1.512 km	m31
pc32	<i>mobile</i>	<i>Police Support</i>	SM	55	85km/h	1.512 km	m32
pc33	<i>mobile</i>	<i>Police Support</i>	SM	59	140km/h	1.512 km	m33
Waverley Police Station (WP)	<i>facility</i>	<i>police station</i>	none	0	0	2.091 km	f4
pc41	<i>mobile</i>	<i>Police Support</i>	WP	60	105km/h	2.091 km	m41
pc42	<i>mobile</i>	<i>Police Support</i>	WP	65	135km/h	2.091 km	m42
pc43	<i>mobile</i>	<i>Police Support</i>	WP	70	145km/h	2.091 km	m43
Eastern Suburbs Private Hospital (ES)	<i>facility</i>	<i>Hospitalisation</i>	none	80	0	2.288 km	f5
am51	<i>mobile</i>	<i>Ambulance Transport</i>	ES	90	95km/h	2.288 km	m51
am52	<i>mobile</i>	<i>Ambulance Transport</i>	ES	80	90km/h	2.288 km	m52
am53	<i>mobile</i>	<i>Ambulance Transport</i>	ES	70	85km/h	2.288 km	m53
The Sydney Clinic (SC)	<i>facility</i>	<i>Hospitalisation</i>	none	40	0	2.999 km	f6
am61	<i>mobile</i>	<i>Ambulance Transport</i>	SC	60	80km/h	2.999 km	m61
am62	<i>mobile</i>	<i>Ambulance Transport</i>	SC	50	70km/h	2.999 km	m62
am63	<i>mobile</i>	<i>Ambulance Transport</i>	SC	40	60km/h	2.999 km	m63

### 3.5.2 Case Study Results and Analysis

The results of case study are shown in Figure 3.6, Figure 3.7, Figure 3.8 and Table 3.8.

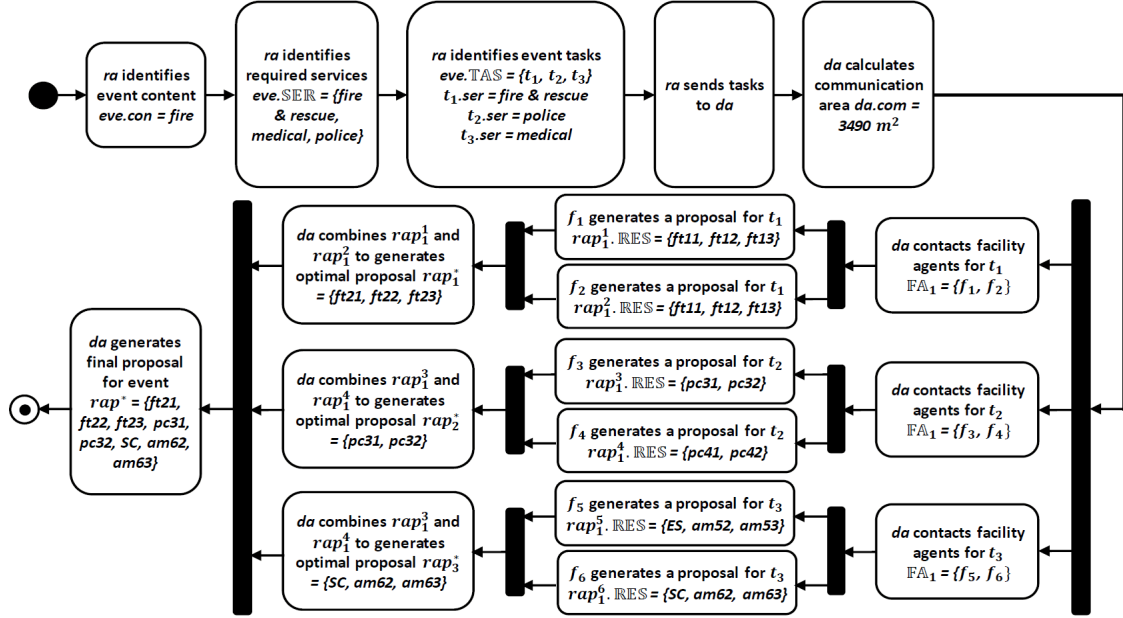


Figure 3.6: The Resource Deployment Activity Diagram for Severity 1 Incident

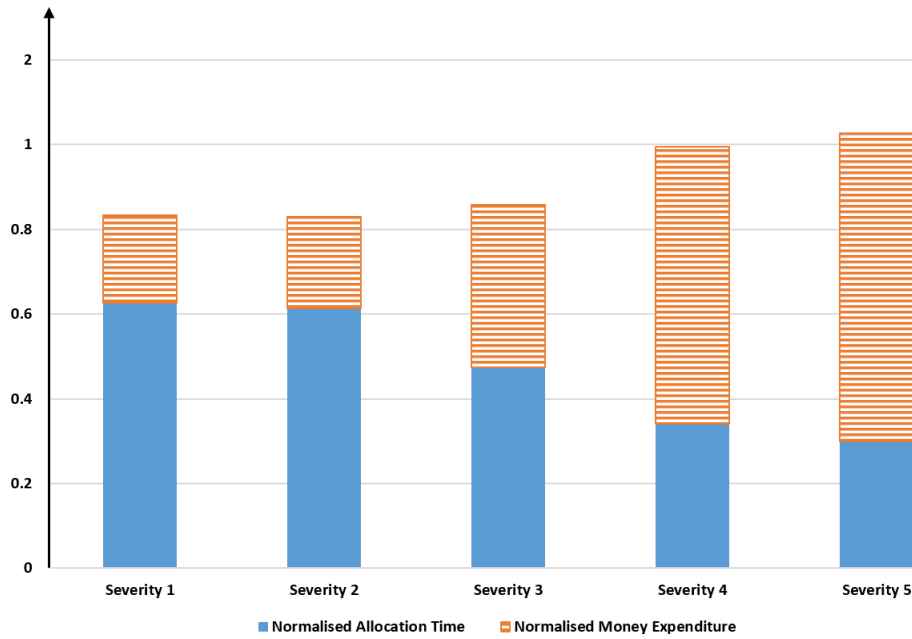
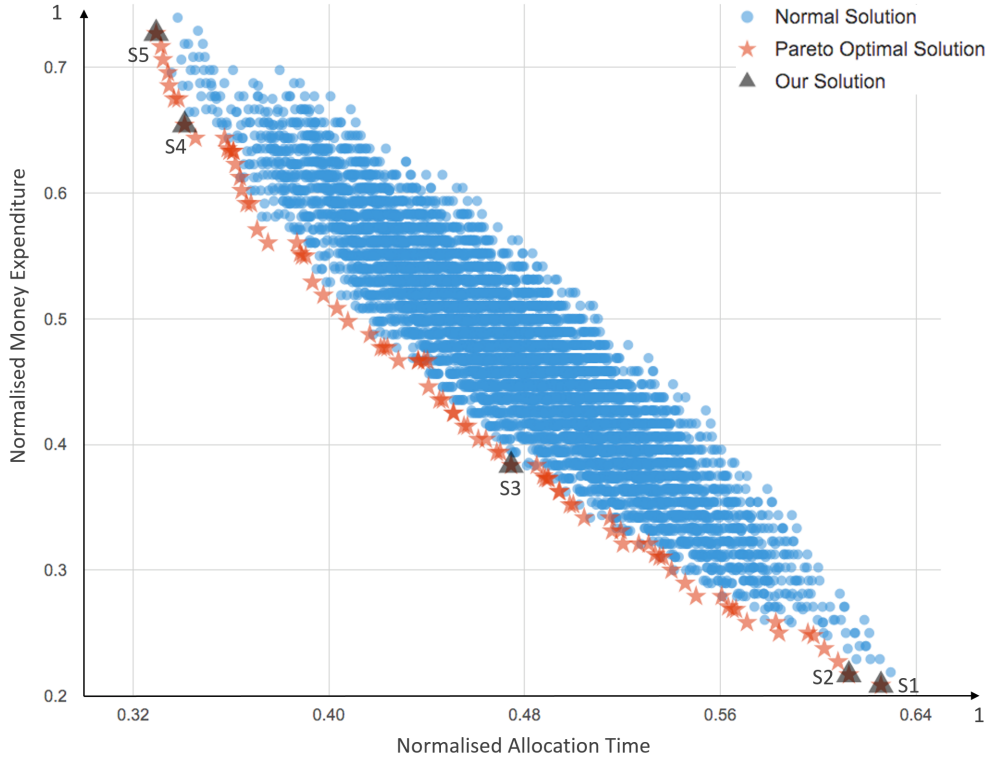


Figure 3.7: Case Study Results

**Figure 3.8:** Pareto Optimality Test**Table 3.8:** The Optimal Plan for each Severity's Incident

Severity	The Resource Name in the Optimal Plan							
1	[ft21]	[ft22]	[ft23]	[pc31]	[pc32]	[SC]	[am62]	[am63]
2	[ft21]	[ft22]	[ft23]	[pc31]	[pc33]	[SC]	[am62]	[am63]
3	[ft12]	[ft13]	[ft22]	[pc31]	[pc33]	[SC]	[am53]	[am63]
4	[ft11]	[ft12]	[ft13]	[pc32]	[pc33]	[ES]	[am52]	[am53]
5	[ft11]	[ft12]	[ft13]	[pc33]	[pc43]	[ES]	[am51]	[am52]

In detail, Figure 3.6 depicts how agents performed actions during the resource deployment for the Severity 1 emergency incident. Figure 3.7 demonstrates the normalised results of unweighted resource deployment time and money expenditure of the same emergency incident with five different severity levels. Figure 3.8 shows the result of the Pareto optimality test. Table 3.8 shows the resources that had been selected by the proposed approach as the optimal plan for each level's emergency incident.

From Figure 3.6, we can see that when the emergency incident happened, a response agent *ra* first identified that the incident content was *fire* and required three emergency services (*fire & rescue, police and medical*). Based on these services in-

formation, *ra* identified three tasks ( $t_1, t_2$  and  $t_3$ ) and their required resources, then these tasks were sent to a deployment agent *da*. *da* calculated its initial communication range according to the resource requirements of the tasks, which was  $3490\text{ m}^2$ , then *da* contacted relevant facility agents and informed them the information of the tasks. More specifically,  $t_1$  was sent to facility Fire Brigades-NSW and Alexandria Fire Station,  $t_2$  was sent to Sydney Mounted Police Stables and Waverley Police Station, and  $t_3$  was sent to Eastern Suburbs Private Hospital and the Sydney Clinic. After receiving the information of these tasks, each facility agent used domain transportation method to select required resources from its controlled resource pool. Since the severity level of this emergency incident is 1, the resources that have been selected by the facility agents were the resources with the lowest money expenditure. Finally, each facility agents submitted their plans to *da* and *da* used domain transportation to combine these plans to generate a final plan for each task.

Figure 3.7 further demonstrates that when the severity of the emergency incident equals to 1, the percentage of the cost generated by resource deployment time is much higher than that of generated by the resource money expenditure (74% versus 26%). These results are reasonable since when an emergency incident has a low severity level, the resource money expenditure is much more important than that of resource deployment time in the cost function (refer to Equations 3.1 and 3.2). Under such a situation, the proposed approach preferentially considers choosing resources with low money expenditures to decrease the final resource deployment cost. From Figure 3.7 we can see that, with the increment of the incident's severity, the cost generated by the resource money expenditure has increased and the resource deployment time has started to generate less and less cost. Besides, we can also find that the combined deployment cost of the incident rises moderately from severity 1 to 5. When the incident comes to the severity 5, the resource deployment time only occupies 22% of the combined cost, while the resource money expenditure occupies 78% of the combined cost. This is mainly caused by the reason that for high severity incidents,

the resource deployment time becomes more important in the cost function, so the proposed approach tried to select resources with high moving velocity and closer distance to the location of the emergency incident. Although high-velocity resources usually come with higher money expenditures and can increase the total cost slightly, however, in high severity incidents, it is no doubt that time is the top propriety we need to consider during the resource deployment process. For example, in Table 3.8, we can see that the fire engines chosen to handle the emergency incident with the severity level 5 are all from the facility resource 'Fire Brigades-NSW', which can provide fire engines with the fast speed but high money expenditure to achieve efficient resource deployment.

Regarding the optimality of the solutions, Figure 3.8 indicates that the proposed approach can achieve the Pareto efficiency [Bre89] for the incident in all five severity levels. More precisely, in Figure 3.8, each point represents the normalised deployment time and money expenditure of a feasible resource deployment solution for the case study incident. A circle point represents a normal solution or a dominated solution in the design objective space, for which there always exists a point in the solution space that is better than this circle point in both resource deployment time and money expenditure. The star points represent a set of Pareto optimal solutions for the case study, which are also called Pareto frontier. Generally, the Pareto optimal solutions are those solutions that any improvement in one objective will result in the worsening of at least one other objective. The five triangle points (i.e. S1, S2, S3, S4 and S5) represent our solutions for the incident in the five severity levels. Apparently, the all five solutions generated by the proposed approach are overlap with the Pareto frontier, which indicates the proposed approach can achieve the optimal solution for the incident in each severity level in terms of deployment time and money expenditure.

### 3.6 Summary

In this chapter, agent-based decentralised resource deployment approach with a comprehensive mathematical model was proposed to handle an emergency incident in metropolitan regions, which provided solutions for achieving research **Objectives 1 and 2**, set up in Section 1.4. In order to efficiently search for an optimal solution for an incident that requires multiple resources with different characteristics, the proposed approach first created a set of tasks based on the emergency services required by the incident. Then, domain transport theory was used to find out the optimal resource mapping from available resources in an environment to the required resources of a task. The proposed approach is capable of handling multiple resource deployment tasks simultaneously and can be used for emergency incidents in different domains by simply adjusting the cost attributes in the cost function.

To increase the practicability of the proposed resource deployment approach, a multi-agent system was designed in a decentralised manner, in which resources were managed by multiple facility agents and mobile agents, distributed in different locations. A response agent was also used to analyse an emergency incident for creating resource deployment tasks. A deployment agent was used to generate the final resource deployment plan for a task based on the plans provided by facility agents. The experimental results and a case study were given to demonstrate the good performance of the proposed approach in terms of resource deployment cost comparing with other centralised and decentralised approaches.

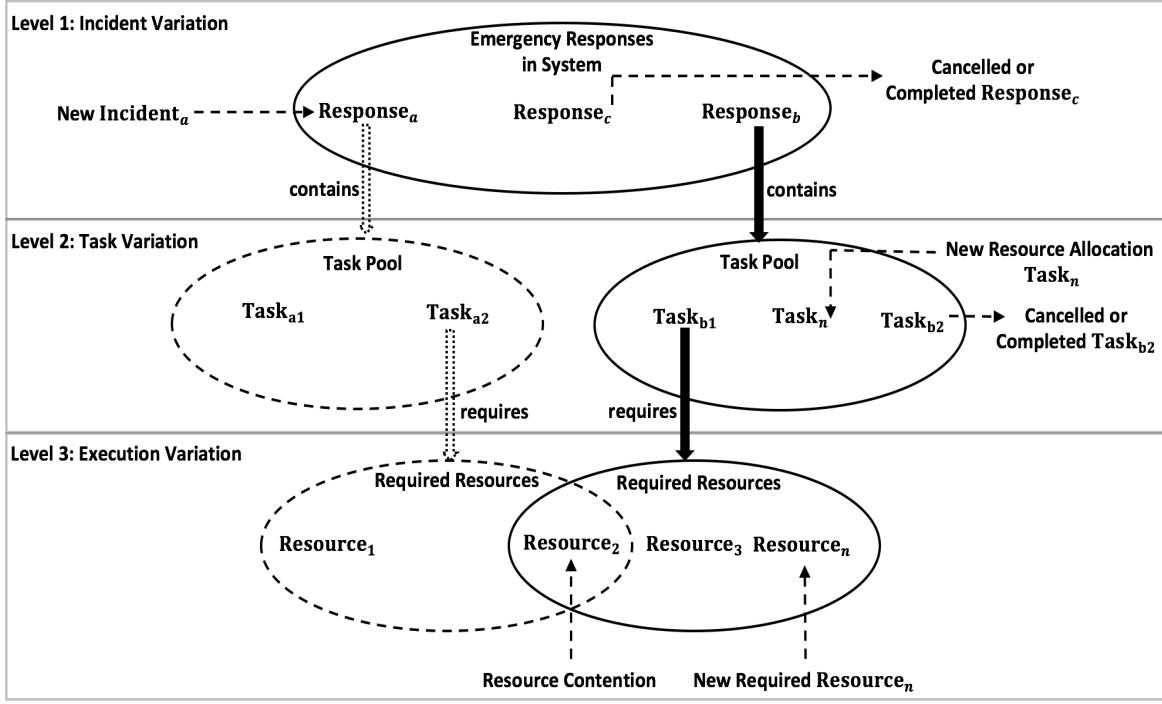


## Chapter 4

# An Agent-Based Dynamic Resource Deployment Approach for Concurrent Incidents

In metropolitan regions, emergency incidents with different severities and demands could happen concurrently at multiple locations, which require rescue resources to be deployed to these incidents simultaneously to handle their emergency responses. In this chapter, an agent-based resource deployment approach is proposed to help emergency departments to effectively generate resources deployment plans to concurrent emergency incidents, which is designed based on the single incident resource deployment approach introduced in Chapter 3.

During the multi-incident resources deployment process, the dynamic features in emergency response could cause different variations, which can be classified into three levels (see Fig 4.1). The first level is the incident variation, referring to that the occurrence of emergency incidents in metropolitan regions could be dynamic and unpredictable. The second level is the task variation, referring to that the total number of the resource deployment tasks and the detail of task requirements in an emergency incident (response) could change dynamically as time passed. The third level is the execution variation, referring to that the execution of a resource deployment could change dynamically due to problems such as traffic jams or resource contention. To ensure the success of rescue operations, it is important for emer-



**Figure 4.1:** The Three Levels of Dynamic Features in Emergency Response

gency response services to automatically adjust and update resource deployment plans in real time situations to adapt variations that caused by dynamic changes in emergency resource deployment.

The proposed multi-incident resource deployment approach contains three mechanisms to dynamically adjust the resource deployment plans based on the changes in terms of incident variation, task variation and execution variation, which provides solutions for solving research **Issues 3** and **4**, identified in Section 1.4. In the experiment, an emergency resource deployment simulation system based on GoogleMaps is developed for testing the proposed approach along with other state-of-the-art dynamic resource deployment strategies in a virtual metropolitan environment.

The rest of this chapter is organised as follows. Section 4.1 gives definitions in the new resource deployment model and the description of multi-incident dynamic resource deployment problems. Section 4.2 introduces the basic principle of agent-based resource deployment for concurrent emergency incidents. Section 4.3 explains the dynamic resource deployment mechanisms for handling the three levels of variations in emergency response. Section 4.4 demonstrates the experimental results and

analysis. Section 5.6 summarises this chapter.

## 4.1 Definitions and Problem Description

This section introduces a new resource deployment model based on the model proposed in Subsection 3.1.1, which contains simplified definitions and terms that are preferable for the resource deployment to the emergency responses of concurrent incidents. Furthermore, the dynamic problems caused by incident variation, task variation and execution are also explained in this section.

### 4.1.1 Definitions of Domain Knowledge

In the proposed multi-incident resource deployment approach, the *environment* of a city is defined as an undirected graph  $G = [\mathbb{V}, \mathbb{Z}]$ , where  $\mathbb{V}$  denotes a set of important locations in the city, and  $\mathbb{Z}$  denotes a set of paths between the important locations. In environment  $G$ , multiple emergency incidents could happen concurrently at different locations with different resource requirements. For each incident, an emergency response that contains a set of resource deployment tasks is generated to handle its requirements. The details of an emergency response and a resource are given in Definition 11 and Definition 12, respectively.

**Definition 11** (Emergency Response). An *emergency response*  $e$  is defined as a four-tuple,  $e = (con, loc, sev, \mathbb{T})$ , where

- $con$  denotes the content type of emergency response  $e$ , such as vehicle accident, house fire, medical emergency, etc.
- $sev \in [1, 5]$  denotes the severity level of emergency response  $e$ , in which  $sev = 1$  indicates the lowest severity and  $sev = 5$  indicates the highest severity.
- $\mathbb{T} = \{t_1, t_2, \dots, t_y\}$  denotes a set of resource deployment tasks to be completed for emergency response  $e$ .

Besides, a resource deployment task  $t \in e.\mathbb{T}$  is further defined as a three-tuple,  $t = (dl, ser, \mathbb{R})$ , where

- $dl$  denotes the deadline of task  $t$ .
- $ser$  denotes the type of emergency service that task  $t$  demands for the emergency resource deployment (i.e. police, medical and fire & rescue).
- $\mathbb{R} = \{r_1, r_2, \dots, r_n\}$  denotes a set of required resources.

**Definition 12** (Resource). A *resource*  $r$  is defined as a six-tuple,  $r = (typ, ser, loc, t, vel, exp)$ , where

- $typ \in \{facility, mobile\}$  denotes the type of resource  $r$ , where *facility* refers to unmovable resources such as hospital, police station, etc; while *mobile* refers to rescue vehicles and personnel.
- $ser$  denotes the type of emergency service that resource  $r$  can provide.
- $loc$  denotes resource  $r$ 's location on environment  $G$ .
- $t$  denotes resource  $r$ 's current task, where  $t = 0$  indicates that  $r$  has no task.
- $vel$  denotes resource  $r$ 's velocity (km/h) and  $exp$  denotes the money expenditure in per hour of resource  $r$ .

To handle an emergency response's tasks, a resource deployment plan needs to be proposed, which is formally defined by Definition 13.

**Definition 13** (Resource deployment Plan). A *resource deployment plan*  $p$  for an emergency response is defined as a two-tuple,  $p = (e, \mathbb{R})$ , where  $\mathbb{R}$  denotes the proposed resources for emergency response  $e$ .

The details of the cost function for a resource and an emergency response are formally defined by Definition 14 and Definition 15, respectively.

**Definition 14** (Resource Cost Function). The *cost function* for a resource  $r$ 's deployment to a task  $e.t_y$  is defined by Equation 4.1:

$$CR(e.t_y, r) = \begin{cases} E^m(r.exp), & \text{if } r.typ = facility \\ (w^t E^t(\frac{DS(r.loc, e.loc)}{r.vel}) + w^m E^m(r.exp)) \times \\ DL(e.t_y, r), & \text{if } r.typ = mobile, \end{cases} \quad (4.1)$$

where function  $DS$  calculates the length of a route between two locations. Evaluation functions  $E^t$  and  $E^m$  convert the values of a resource's deployment time and money expenditure to normalised values in-between 0 and 1, respectively.  $w^t$  and  $w^m$  represent the weighting of the resource's deployment time and money expenditure, respectively, which are determined by Equation 4.2:

$$\begin{cases} w^t = \frac{e.sev}{5} \\ w^m = 1 - \frac{e.sev}{5}. \end{cases} \quad (4.2)$$

Furthermore, function  $DL$  evaluates whether resource  $r$  can be delivered before task  $e.t_y$ 's deadline, which is defined by Equation 4.3:

$$DL(e.t_y, r) = \begin{cases} 1, & \text{if } \frac{DS(r.loc, e.loc)}{r.vel} \leq e.t_y.dl \\ +\infty, & \text{if } \frac{DS(r.loc, e.loc)}{r.vel} > e.t_y.dl. \end{cases} \quad (4.3)$$

**Definition 15** (Emergency Response Cost Function). The cost function for an emergency response  $e$  is defined by Equations 4.4:

$$CE(e, p) = \sum_{e.t_y \in e.\mathbb{T}} \sum_{r_n \in p_y.\mathbb{R}} CR(e.t_y, r_n) \text{ subject to } p_y.\mathbb{R} \in p.\mathbb{R}. \quad (4.4)$$

### 4.1.2 Problem Description

For a set of concurrent emergency responses  $\mathbb{E}$ , various resource deployment plans with different resource combinations can be proposed to fulfil their tasks. The main problem to be solved in the proposed approach is to search for an optimal plan  $p^*$  for each response  $e \in \mathbb{E}$  to minimise the total resource deployment cost. The objective function  $OA$  for the resource deployment of all concurrent emergency responses in  $\mathbb{E}$  is defined by Equation 4.5:

$$OA = \arg \min_{\mathbb{P}^* = \{p_1^*, \dots, p_x^*\}} \sum_{e_x \in \mathbb{E}} CE(e_x, p_x^*) \text{ subject to } p_y^*.\mathbb{R} \in \mathbb{R}^g, \quad (4.5)$$

where  $p_x^*$  denotes the optimal resource deployment plan that has the minimal resource deployment cost for emergency response  $e_x$  and  $\mathbb{R}^g$  denotes the available resources in environment  $G$ .

As described in research **Issue 3**, identified in Section 1.4, one of the major concern of the resource deployment for concurrent emergency responses is the resource contention between different incidents. In this chapter, a resource content problem is classified as one of the dynamic problems caused by the execution variation. More precisely, during the resource deployment process for emergency responses in  $\mathbb{E}$ , the dynamic nature of emergency management might introduce many unpredictable variations, which could cause different resource deployment problems in the three-level variations (see Figure 4.2). To achieve the optimal resource deployment results that can fulfil objective function  $OA$  (i.e. Equation 4.5), these dynamic problems need to be addressed appropriately.

Variations	Dynamic Resource Deployment Problems	
<b>Level 1: Incident Variation</b>	<b>(1.1) Handling New Incidents (Responses)</b>	<b>(1.2) Handling Cancelled or Completed Responses</b>
<b>Level 2: Task Variation</b>	<b>(2.1) Handling New Resource Deployment Tasks</b> <b>(2.3) Handling New Required Resources in Tasks</b>	<b>(2.2) Handling Cancelled or Completed Tasks</b> <b>(2.4) Addressing Task Deadline Variation</b>
<b>Level 3: Execution Variation</b>	<b>(3.1) Addressing Resource Contention</b>	<b>(3.2) Addressing Resource Arriving Time Variation</b>

**Figure 4.2:** Problem Description

In detail, there are two problems to be considered in the level of incident variation: **(1.1)** how to handle the emergency response for a newly arrived incident and **(1.2)** how to handle a cancelled or completed emergency response. In the incident variation, dynamically recycling the usages of rescue resources from cancelled and completed emergency responses is critical for the proposed approach to generate highly efficient resource deployment plans for newly arrived incidents.

There are four problems to be considered in the level of task variation: **(2.1)** how to handle a newly arrived task of an emergency response; **(2.2)** how to handle a

cancelled or completed task of an emergency response; **(2.3)** how to address the situation when a task's deadline has changed and **(2.4)** how to handle new resource requirements of a task. In the task variation, dynamically adjusting resource deployment plans to adaptive any changes of the tasks in an emergency response is the key to ensure a rescue operation can be conducted successfully.

There are two problems to be considered in the level of execution variation: **(3.1)** how to handle a resource contention problem, when a resource is requested by multiple tasks from different emergency responses at the same time and **(3.2)** how to address the situation when a resource's estimated arrival time has changed due to some unexpected situations such as traffic jam. In the execution variation, dynamically coordinating rescue resources to effectively solve their conflicts is the preconditions for resource deployment plans to be executed.

In summary, the above eight dynamic problems might appear concurrently with each other during the resource deployment process and it is extremely important for a resource deployment approach to possess appropriate mechanisms to efficiently handle these problems in real time. Otherwise, the resource deployment results might not be able to meet the demands of emergency responses and the efficiency of rescue operations for emergency incidents could be affected significantly. The following two sections provide the detail of resource deployment process for the emergency responses for concurrent incidents, and introduce the mechanisms for solving the dynamic problems caused by different variations during the resource deployment process, respectively.

## 4.2 Agent-based Resource Deployment for Concurrent Emergency Incidents

The proposed multi-incident resource deployment approach is also implemented based on the same agent definitions introduced in Subsection 3.3.1. More precisely, to handle the concurrent emergency responses  $\mathbb{E}$  for multiple incidents, the proposed multi-incident approach generates multiple separated processing threads, with each

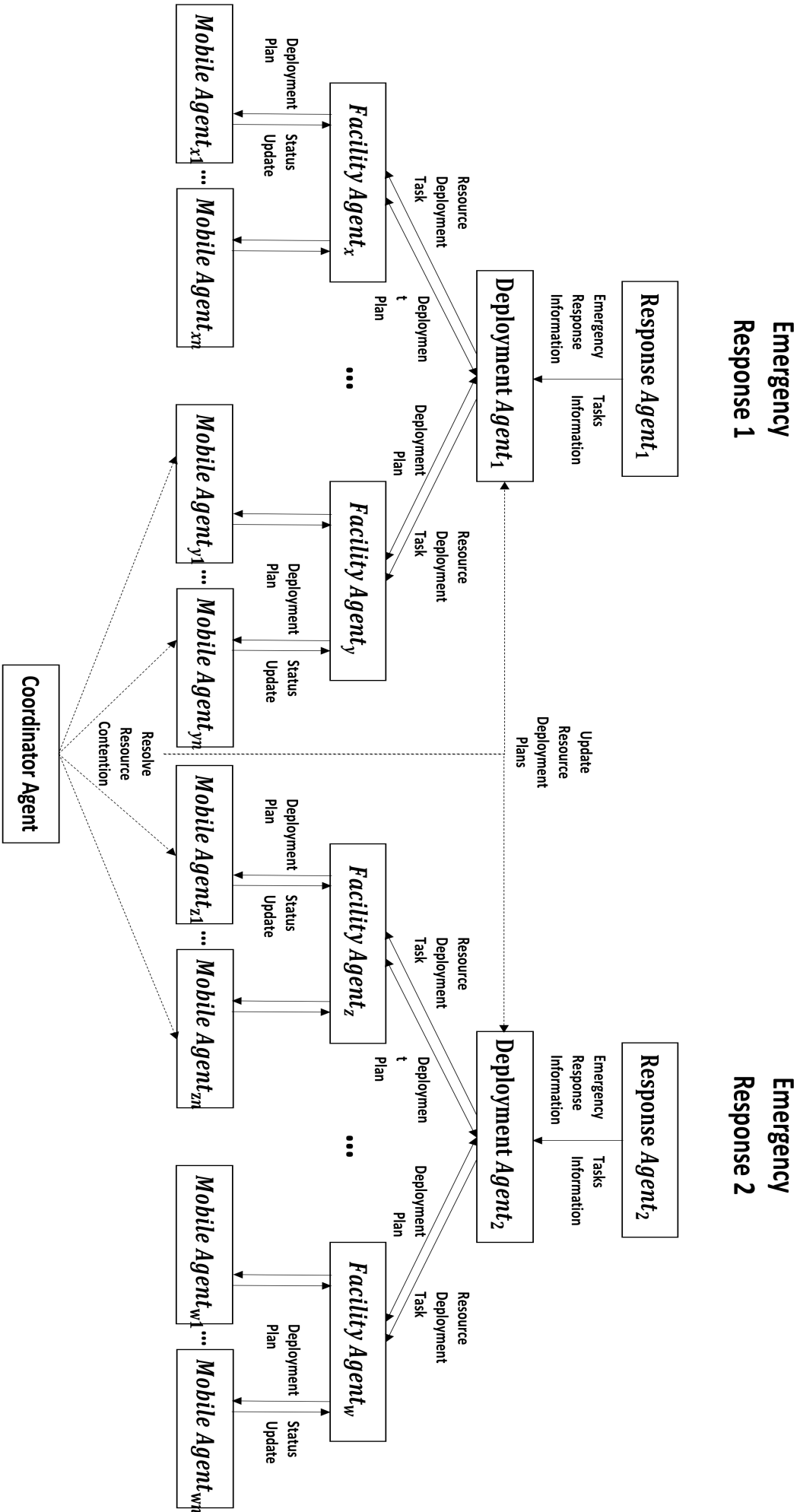
thread responsible for the resource deployment process for a single incident. The resource deployment process for a single incident is similar to the process introduced in Section 3.3.2, which involves multiple *mobile agents*, *facility agents*, one *response agent* and one *deployment agent* (see Algorithm 1 in Section 3.3.2 for detail).

To address the potential resource contention problems caused by the execution variation, a *coordinate agent* is employed in the proposed approach, which is defined by the following definition.

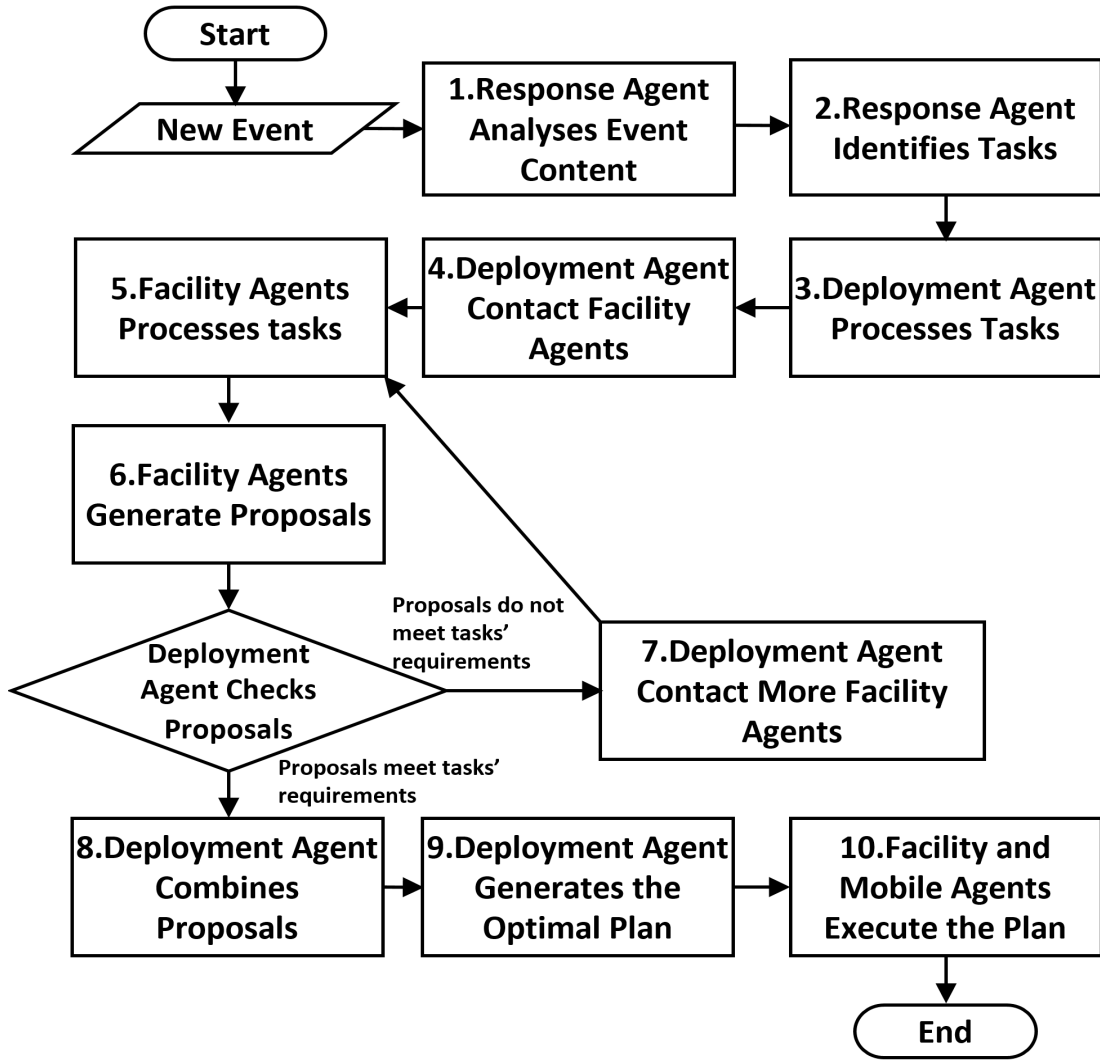
**Definition 16** (Coordinator Agent). A *coordinator agent* is represented by *ca*, which is dynamically generated to address resource contention between the emergency responses for multiple incidents. A coordinator agent has two major functionalities, including **(1)** dynamically reassigning a contention resource to the most appropriate task and finding replacement resources for other tasks; and **(2)** informing relevant deployment agents about the new resource deployment plans.

Figure 4.3 in the next page provides an example of agents' interactions (with a coordinator agent) during the resource deployment process for the emergency responses of two concurrent incidents. The detailed mechanism for addressing a resource contention problem during multi-incident resource deployment will be explained in Subsection 4.3.3.





**Figure 4.3:** An Example of Agents' Interactions for The Emergency Responses of Concurrent Incidents



**Figure 4.4:** An Example of A Resource Deployment Procedure for The Emergency Response of A Single Incident

As described in Subsection 4.1.2, during the resource deployment procedure for concurrent emergency responses  $\mathbb{E}$ , incident variation, task variation and execution variation might cause different dynamic problems. Take Figure 4.4 as an example for a common procedure in resource deployment for the emergency response of a single incident. The problems of incident variation could occur in the Process 1; the problems of task variation could occur in the Processes 1, 2 and 10 and the problems of execution variation could occur in the Processes 9 and 10. Failure to handle any above dynamic problems would likely lead to emergency response delay and rescue operation failure.

### 4.3 Dynamic Resource Deployment Mechanisms

Subsections 4.3.1 - 4.3.3 propose three dynamic resource deployment mechanisms to handle the problems caused by incident variation, task variation and execution variation during the multi-incident resource deployment process.

#### 4.3.1 A Resource Deployment Mechanism for Incident Variation

The first problem of incident variation is to handle a newly arrived incident  $e$  (i.e. **Problem 1.1** in Subsection 4.1.2), which can be effectively addressed by using Algorithm 1, proposed in Section 3.3.2 and a new emergency response  $e$  will be added into the current emergency response set  $\mathbb{E}$ .

The second problem of incident variation is to handle a cancelled or completed emergency response  $e$  of an incident (i.e. **Problem 1.2** in Subsection 4.1.2), in which resources that have been assigned to the emergency response  $e$  need to be released effectively, thus they could be used for existing or incoming incidents to further improve the efficiency of emergency responses. The resource releasing process for a cancelled or completed emergency response  $e$  is formally described by Algorithm 2.

---

**Algorithm 2** : Resource Releasing Process for a Cancelled & Completed Incident

---

**Input:**  $e$

- 1:  $ra$  detects the cancel or the completion of  $e$
  - 2:  $ra$  informs  $da$  to releasing resources from  $e$
  - 3: **for all**  $fa$  involved in the resource deployment of  $e$  **do**
  - 4:      $da$  informs  $fa$  to cancel execution
  - 5:     **for all**  $ma \in fa.MA$  involves in the resource deployment of  $e$  **do**
  - 6:          $fa$  informs  $ma$  to cancel execution
  - 7:          $ma$  releases its resource usage
  - 8:     **end for**
  - 9: **end for**
  - 10: release the usage of  $da$  and  $ra$
  - 11: update  $\mathbb{E} \leftarrow \mathbb{E} \setminus \{e\}$
- 

In the above algorithm, after a deployment agent  $da$  receives the resource releasing

orders from a response agent  $ra$ , deployment agent  $da$  informs relevant facility agents and mobile agents to release the usages of their resources for emergency response  $e$  (Lines 2-9). Then, deployment agent  $da$  and response agent  $ra$  release their duties of handling emergency response  $e$  (Line 10).

### 4.3.2 A Resource Deployment Mechanism for Task Variation

The first problem of task variation is to handle new tasks of an emergency response (i.e. **Problem 2.1** in Subsection 4.1.2). For example, the emergency response of a small house fire may only have one task that requires fire & rescue service at first. But when the fire has spreaded and people get burned by the fire, a new task that requires medical service needs to be added to the emergency response. Apparently, failure to automatically update the resource deployment plan for this emergency response can cause the lost of people's life. The resource deployment process for a new added task  $t$  in emergency response  $e$  is formally described by Algorithm 3.

---

**Algorithm 3** : Resource Management Process for a New Task

---

**Input:**  $e$

- 1:  $ra$  detects new requirements from  $e$
  - 2:  $ra$  identifies a new task  $t$
  - 3:  $ra$  updates  $e.\mathbb{T} \leftarrow e.\mathbb{T} \cup \{t\}$
  - 4:  $ra$  sends  $t$  to  $da$
  - 5: the process goes to Line 6 of Algorithm 1
- 

The second problem of task variation is to handle a cancelled or completed task  $t_y$  in emergency response  $e$  (i.e. **Problem 2.2** in Subsection 4.1.2), which has the similar solution to the problem of releasing resources for a cancelled or completed emergency response (i.e. Algorithm 2).

The third problem of task variation is the changing of a task  $t_y$ 's deadline in an emergency response  $e$  (i.e. **Problem 2.3** in the Subsection 4.1.2). In real life situations, this problem could cause by the severity increase of an emergency response. For example, a medical emergency could change from a non-life-threatening situ-

ation to a life-threatening situation due to the deterioration of a patient's health condition, which requires immediate ambulance transportation with shorter task deadline comparing to the original situation. Incapable of adjusting a resource deployment plan based on the task deadline variation might directly cause the failure of an emergency response. In detail, if task  $t_y$  has an on-execution plan  $p_y$ , the deployment agent needs to ensure that each of the resource  $r$  in plan  $p_y$  can still arrive the incident location  $e.loc$  on time, which can be checked by using Equation 4.6:

$$RT(r, e) = \begin{cases} 0, & \text{if } \frac{DS(r.loc, e.loc)}{r.vel} \leq e.t_y.dl \\ 1, & \text{if } \frac{DS(r.loc, e.loc)}{r.vel} > e.t_y.dl, \end{cases} \quad (4.6)$$

where  $RT = 0$  indicates that resource  $r$  can still arrive the incident scene before the task deadline  $e.t.dl$ , while  $RT = 1$  indicates that  $r$  cannot arrive the incident scene on time.

For a resource  $r$  that cannot arrive the incident location on time, the deployment agent of task  $e.t_y$  will first try to locate an optimal replacement resources  $r^r$  from the candidate resource pool  $\mathbb{R}_y^c$  of task  $e.t_y$  to substitute  $r$ . However, if such a resource cannot be found, the deployment agent will try to negotiate with the response agent to modify the task deadline. The resource deployment process for the task deadline variation is formally described by Algorithm 4.

---

**Algorithm 4** : Resource Deployment Process for Task Deadline Variation

---

**Input:**  $e$ 

```

1:  $ra$  detects variation of  $e$ 
2:  $ra$  updates  $e.t_y.dl \leftarrow e.t_y.dl'$ 
3:  $ra$  informs  $da$  the change of  $e.t_y.dl$ 
4:  $da$  acquires  $\mathbb{R}_y^c$  and  $p_y$  of  $e.t_y$ 
5: for all  $r \in p_y.\mathbb{R}$  do
6:   if  $RT(r, e)$  then
7:      $da$  creates  $r^r = 0$ 
8:      $r^r \leftarrow \arg \min_{r \in \mathbb{R}_y^c} CR(e.t_y, r)$  subject to
9:        $r \neq r^c \wedge r.ser = r^c.ser$ 
10:    if  $r^r \neq 0$  then
11:      update  $p_y.\mathbb{R} \leftarrow (p_y.\mathbb{R} \setminus \{r\}) \cup \{r^r\}$ 
12:      update  $r^r.t \leftarrow e.t_y$ 
13:       $da$  notifies relevant  $fa$  to send out  $r^r$ 
14:       $da$  notifies relevant  $fa$  to release  $r$ 's usage
15:    else
16:       $da$  negotiates with  $ra$  to change  $e.t_y.dl \leftarrow \frac{DS(r.loc, e.loc)}{r.vel}$ 
17:      if negotiation fails then
18:         $da$  reports  $e.t_y$  fail to complete
19:      else
20:         $da$  reports no need to change  $r$ 
21:      end if
22:    end if
23:  else
24:     $da$  reports no need to change  $r$ 
25:  end if
26: end for

```

---

The Algorithm 4 is explained as follows. After a response agent  $ra$  detects the task deadline variation of task  $t_y$  in emergency response  $e$ ,  $ra$  will inform relevant deployment agent  $da$  to check whether the original resource deployment plan  $p_y$  is still a feasible plan (Lines 1-6). If the original plan  $p_y$  is compromised, deployment agent  $da$  needs to find suitable replacement resources to meet the requirement of the new task deadline and updates the old plan  $p_y$  (Lines 7-13). Nevertheless, if such replacement resources cannot be found, deployment agent  $da$  needs to negotiate with

resource agent  $ra$  regarding the new task deadline and decide whether task  $t_y$  can still be completed (Lines 14-17).

The fourth problem of task variation is the resource requirements changing of a task  $t_y$  in an emergency response  $e$ . Take the house fire as an example again. A problem could occur when a small house fire spreads and causes adjacent houses on fire, then more fire engines might need to be added into the original task  $t_y$  to handle the new situation. In the proposed approach, a response agent  $ra$  will continuously monitor emergency response  $e$  and update new resource requirements  $\mathbb{R}'$  to a appropriate task (i.e.  $\mathbb{R}' \cup t_y.\mathbb{R}$ ), then a deployment agent  $da$  will use Algorithm 1 to process the new resource deployment requirements of task  $t_y$  (starting from Line 7 in Algorithm 1).

### 4.3.3 A Resource Deployment Mechanism for Execution Variation

In the level of execution variation, the first problem that needs to be considered is the resource contention (i.e. **Problem 3.1** in Subsection 4.1.2). More precisely, in the proposed approach, a resource contention situation could occur when an available resource is demanded concurrently by multiple tasks from different emergency responses or a resource that has already been deployed to a task required by newly arrived emergency responses in a metropolitan region. Failure to address a resource contention situation will result in failure to execute some resource deployment plans.

For a set of resource contention tasks, resource deployment processes are interrelated with each other. To minimise the total deployment cost of these interrelated tasks and fulfil objective function  $OA$  (i.e. Equation 4.5), Algorithm 5 is introduced to handle the contention of a resource. The algorithm will reassign this resource to an appropriate by considering three factors, which are (1) the cost that has already been spent on this resource, (2) the cost of finding replacement resources for other tasks and (3) the future cost of using this resource to complete a task. Formally, let  $r^c$  denote a resource, required by a set of tasks  $\mathbb{T}^c = \{e_1.t_1, \dots, e_x.t_y\}$  at the same

time;  $\mathbb{R}_{xy}^c$  denotes a set of candidate resources that can be deployed to  $e_x.t_y \in \mathbb{T}^c$  and  $\mathbb{P}^c = \{p_{11}, \dots, p_{xy} | p_{xy} \cdot \mathbb{R} \subset \mathbb{R}_{xy}^c\}$  denotes a set of plans that are sent out for execution for the tasks in  $\mathbb{T}^c$ . When a resource contention happens, the conflict resource  $r^c$  needs to be reassigned to an appropriate task in  $\mathbb{T}^c$  and extra replacement resources need to be located for other tasks to minimise the overall deployment cost of all involved resource contention tasks. The objective function  $OR$  for the reassignment of conflict resource  $r^c$  is formally defined by Equation 4.7:

$$OR = \arg \min \begin{cases} e^*.t^* \in \mathbb{T}^c, \\ (e_1.t_1, r_{11}^r) \in (\mathbb{T}^u, \mathbb{R}^o), \\ \dots, \\ (e_x.t_y, r_{xy}^r) \in (\mathbb{T}^u, \mathbb{R}^o) \end{cases} \quad (4.7)$$

$$CR(e^*.t^*, r^c) + \sum_{e_x.t_y \in \mathbb{T}^u} CR(e_x.t_y, r_{xy}^r),$$

where  $\mathbb{R}^o$  denotes a common resource pool that is generated by the combination of all candidate resource pools of tasks in  $\mathbb{T}^c$ ;  $e^*.t^*$  denotes the winning task that has gotten the conflict resource  $r^c$ ;  $\mathbb{T}^u = \{\mathbb{T}^c \setminus e^*.t^*\}$  denotes a set of un-winning tasks that did not get  $r^c$  and order pair  $(e_x.t_y, r_{xy}^r)$  denotes an un-winning task  $e_x.t_y$  and a replacement resource  $r_{xy}^r$  that  $e_x.t_y$  selects from the common resource pool  $\mathbb{R}^o$  to substitute missing  $r^c$ .

In order to fulfil  $OR$ , the proposed approach dynamically generates a coordinator agent to handle the resource reassignment process, which is represented by  $ca$ . The coordinator agent  $ca$  first needs to identify an optimal replacement resources  $r_{xy}^r$  for each task  $e_x.t_y \in \mathbb{T}^c$ .  $ca$  uses recursion to handle any further resource contention, until all tasks in  $\mathbb{T}^c$  have found an unique replacement resources  $r_{xy}^r$ . Then,  $ca$  needs to select the winning task  $e^*.t^*$  for  $r^c$ , which is calculated by Equation 4.8:

$$e^*.t^* = \arg \min_{e_x.t_y \in \mathbb{T}^c} \frac{CR(e_x.t_y, r^c)' + CR(e_x.t_y, r^c)}{CR(e_x.t_y, r_{xy}^r)}, \quad (4.8)$$

where  $CR(e_x.t_y, r^c)'$  denotes the cost that has already been spent on resource  $r^c$  for task  $e_x.t_y$ ;  $CR(e_x.t_y, r^c)$  denotes the current cost of using resource  $r^c$  to complete



task  $e_x.t_y$  according to  $r^c$ 's current position in the environment  $G$  and  $CR(e_x.t_y, r_{xy}^r)$  denotes the cost of using the best replacement resource  $r_{xy}^r$  to complete task  $e_x.t_y$ .

The resource reassignment process for a resource  $r^c$  that is required by a set of task  $\mathbb{T}^c$  is formally described by Algorithm 5.

---

**Algorithm 5** : Resource\_Reassignment Process

---

**Input:**  $r^c, \mathbb{T}^c, \mathbb{R}^o, \mathbb{P}^c$

**Require:**  $|\mathbb{R}^o| > |\mathbb{T}^c|$

```

1: for all  $e_x.t_y \in \mathbb{T}^c$  do
2:    $r_{xy}^r \leftarrow \arg \min_{r \in \mathbb{R}^o} CR(e_x.t_y, r)$  subject to
      $r \neq r^c \wedge r.ser = r^c.ser$ 
3:    $r_{xy}^r.\mathbb{T} \leftarrow r_{xy}^r.\mathbb{T} \cup \{e_x.t_y\}$ 
4: end for
5: for all  $e_x.t_y \in \mathbb{T}^c$  do
6:   if  $|r_{xy}^r.\mathbb{T}| > 1$  then
7:     Algorithm 5 ( $r_{xy}^r, r_{xy}^r.\mathbb{T}, \mathbb{R}^o, \mathbb{P}^c$ )
8:   end if
9: end for
10:  $e^*.t^* \leftarrow \arg \min_{e_x.t_y \in \mathbb{T}^c} \frac{CR(e_x.t_y, r^c)' + CR(e_x.t_y, r^c)}{CR(e_x.t_y, r_{xy}^r)}$ 
11: update  $r^c.t \leftarrow e^*.t^*$ 
12:  $\mathbb{R}^o \leftarrow \mathbb{R}^o \setminus \{r^c\}$ 
13:  $\mathbb{T}^u \leftarrow \mathbb{T}^c \setminus \{e^*.t^*\}$ 
14: for all  $e_x.t_y \in \mathbb{T}^u$  do
15:   update  $r_{xy}^r.t \leftarrow e_x.t_y$ 
16:   update  $p_{xy}.\mathbb{R} \leftarrow (p_{xy}.\mathbb{R} \setminus \{r^c\}) \cup \{r_{xy}^r\}$ 
17:    $\mathbb{R}^o \leftarrow \mathbb{R}^o \setminus \{r_{xy}^r\}$ 
18: end for

```

---

In Algorithm 5, after a coordinator agent  $ca$  receives all required information of a resource contention problem (i.e.  $r^c, \mathbb{T}^c, \mathbb{R}^o$  and  $\mathbb{P}^c$ ), coordinator agent  $ca$  will search for replacement resources for every resource contention tasks  $\mathbb{T}^c$  (Lines 1-9). Then, Equation 4.8 is used to reassign the conflict resource  $r^c$  to an appropriate task (Lines 10-13). Finally, coordinator agent  $ca$  updates all resource content tasks' execution plans with replacement resources (Lines 14-18).

The second problem of execution variation is the changing of the estimated arrival time of a resource  $r$  (i.e. **Problem 3.2** in Subsection 4.1.2), which could be caused

by traffic jams, mobile resources malfunction, etc. Failure to handle the changes in resource arrival time might result in that a rescue operation cannot be conducted in time, which would increase the dangerous levels of victims. The resource deployment process for handling the resource arrival time variation problem is formally described by Algorithm 6.

---

**Algorithm 6** : Resource deployment Process for the Resource Arrival Time Variation Problem

---

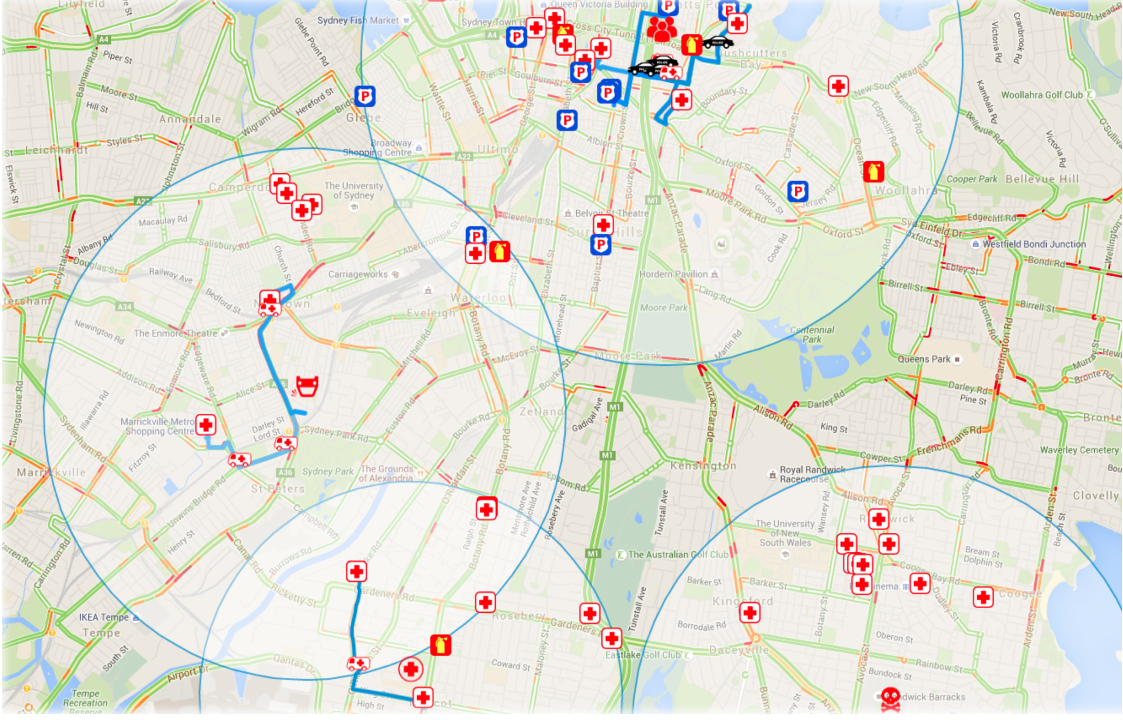
**Input:**  $r, e$

- 1:  $ma$  detects changes of the deliver time of its resource  $r$
  - 2:  $ma$  evaluates  $RT(r, e)$
  - 3: **if**  $RT(r, e)$  **then**
  - 4:      $ma$  informs  $fa$  of changing  $r$
  - 5:      $fa$  informs  $da$  to find a replacement resource for  $r$
  - 6:     the process goes to Algorithm 4 line 7
  - 7: **end if**
- 

Generally, when the estimated arrive time of resource  $r$  has changed, the mobile agent, responsible for the control of resource  $r$ , needs to use Equation 4.6 to evaluate whether resource  $r$  can still arrive its task's location before the task deadline (Lines 1-3) and decide whether a replacement resource needs to be found to ensure the resource deployment plan for the task that can still be fully executed (Lines 4-7).

## 4.4 Experiments and Evaluation

### 4.4.1 Experimental Setting



**Figure 4.5:** Agent-Based Emergency Resource deployment Simulator for Concurrent Incidents

We have conducted an experiment to evaluate the proposed approach in terms of the total resource deployment cost for concurrent incidents and the task completion rate (i.e. all required resources of a task need to arrive before the task’s deadline). We have upgraded our agent-based resource deployment simulator to handle the dynamic resource deployment for concurrent emergency incidents in GoogleMaps (see Figure 4.5). To test the work efficiency of the three proposed dynamic resource deployment mechanisms for different variations in emergency management, the experiment was organised into three scenarios. In each scenario, only certain variations and dynamic problems would occur during the resource deployment process (see Table 4.1 for details).

**Table 4.1:** Scenario Setting

Scenario	Variations	Problems
1	Execution	3.1, 3.2
2	Task, Execution	2.1, 2.2, 2.3, 2.4, 3.1, 3.2
3	Incident, Task, Execution	1.1, 1.2, 2.1, ..., 3.2

The proposed approach was tested along with other three related agent-based emergency resource deployment approaches [GKR11; LIB08; WHM15]. More precisely, in [GKR11], Gabdulkhakova et al. proposed an agent-oriented mathematical model for the resource rationalisation in emergency situations based on resource money expenditure and distance, in which resource execution variation is taken into account during the resource deployment process. López et al [LIB08] proposed an auction-based multi-agent system for coordinating ambulances for emergency medical services with the consideration of task variation. Widener et al. [WHM15] introduced a hybrid agent-based resource deployment approach for the spatial (i.e. distance) optimisation of emergency relief teams given dynamic service demand regarding incident variation and execution variation. The major characteristics of each of these four approaches in terms of variation handling and optimisation objectives are listed in Table 4.2.

**Table 4.2:** Four Testing Approaches

Approach	Variation	Optimisation
[GKR11]	Execution	Money & Distance
[LIB08]	Task	Time
[WHM15]	Incident, Execution	Distance
Our Approach	Incident, Task, Execution	Money & Time

In each scenario, all of four approaches need to deploy resources to 10 concurrent incidents for 10 times, respectively. The task completion ratio and the average total cost of 10 incidents were recorded. The total cost was calculated by Equation 4.4 and the results were normalised between 0 and 1 by their minimal and maximal values. A high value of the normalised result indicates a high cost and vice verse. The cost

for an incomplete task is counted as 1. The experimental parameters of incidents, tasks and resources were randomly generated based on the parameter settings from Tables 4.3, 4.4 and 4.5, respectively.

**Table 4.3:** Parameters for incident's Setting

$con$	$sev$	$ \mathbb{T} $
$\{fire, vehicle\ accident, robbery\}$	$[1-5]$	$[1 - 3]$

**Table 4.4:** Parameters for Task's Setting

No.	$ser$	$dl$	$ \mathbb{R} $	$r.typ$
$[1 - 3]$	$\{fire\ \&\ rescue, medical, police\}$	$[5min - 15min]$	15	$\{mobile, facility\}$

**Table 4.5:** Parameters for Resource's Setting

$typ$	$vel$	$exp$
$mobile$	$[20km/h - 100km/h]$	$[50 - 100]$
$facility$	0	$[50 - 100]$

#### 4.4.2 Result Analysis and Discussions

The results from the three experimental scenarios are shown in Figure 4.6, Figure 4.7 and Figure 4.8, respectively, in the next three pages.

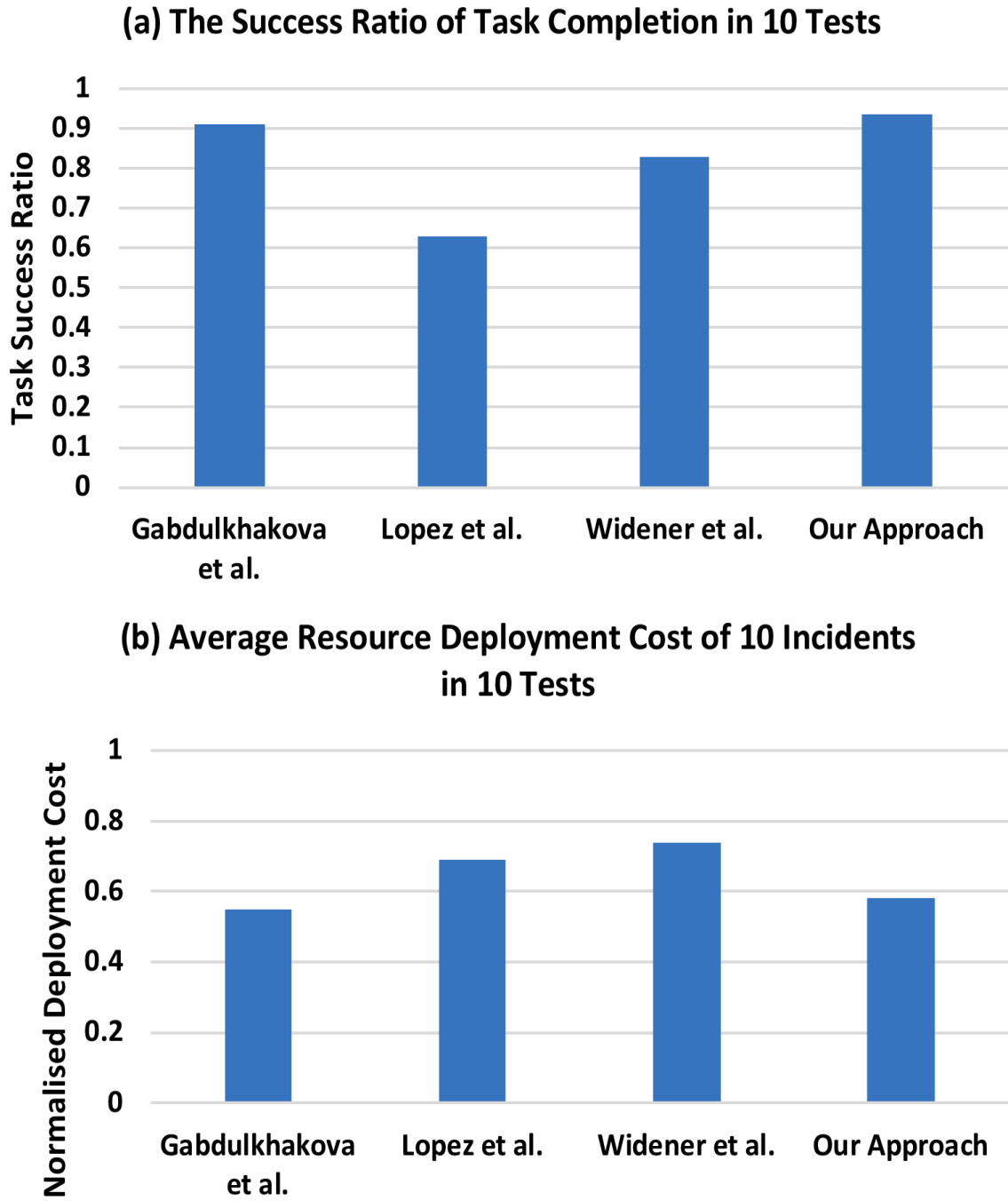


Figure 4.6: The Experimental Results of Scenario 1

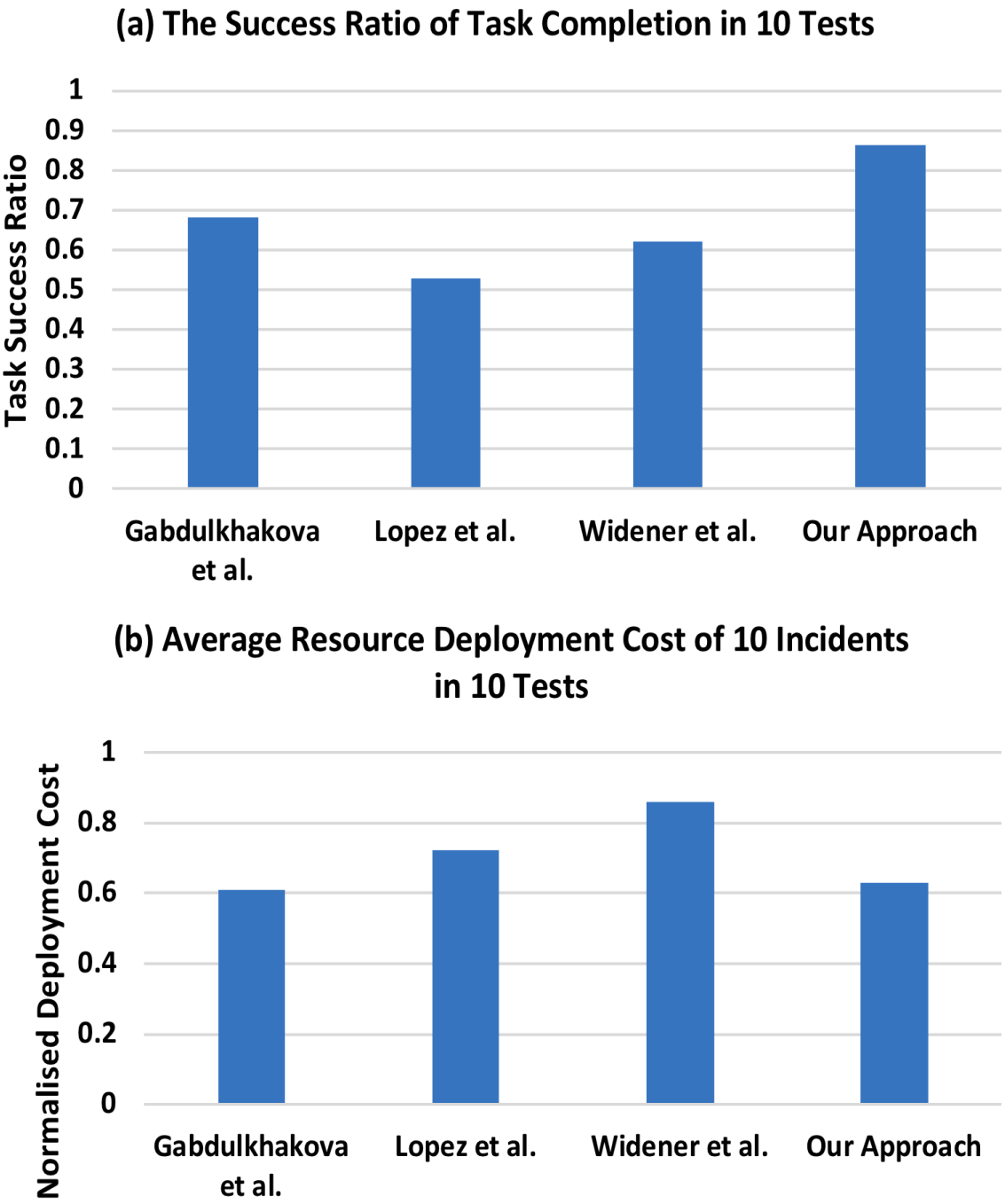


Figure 4.7: The Experimental Results of Scenario 2

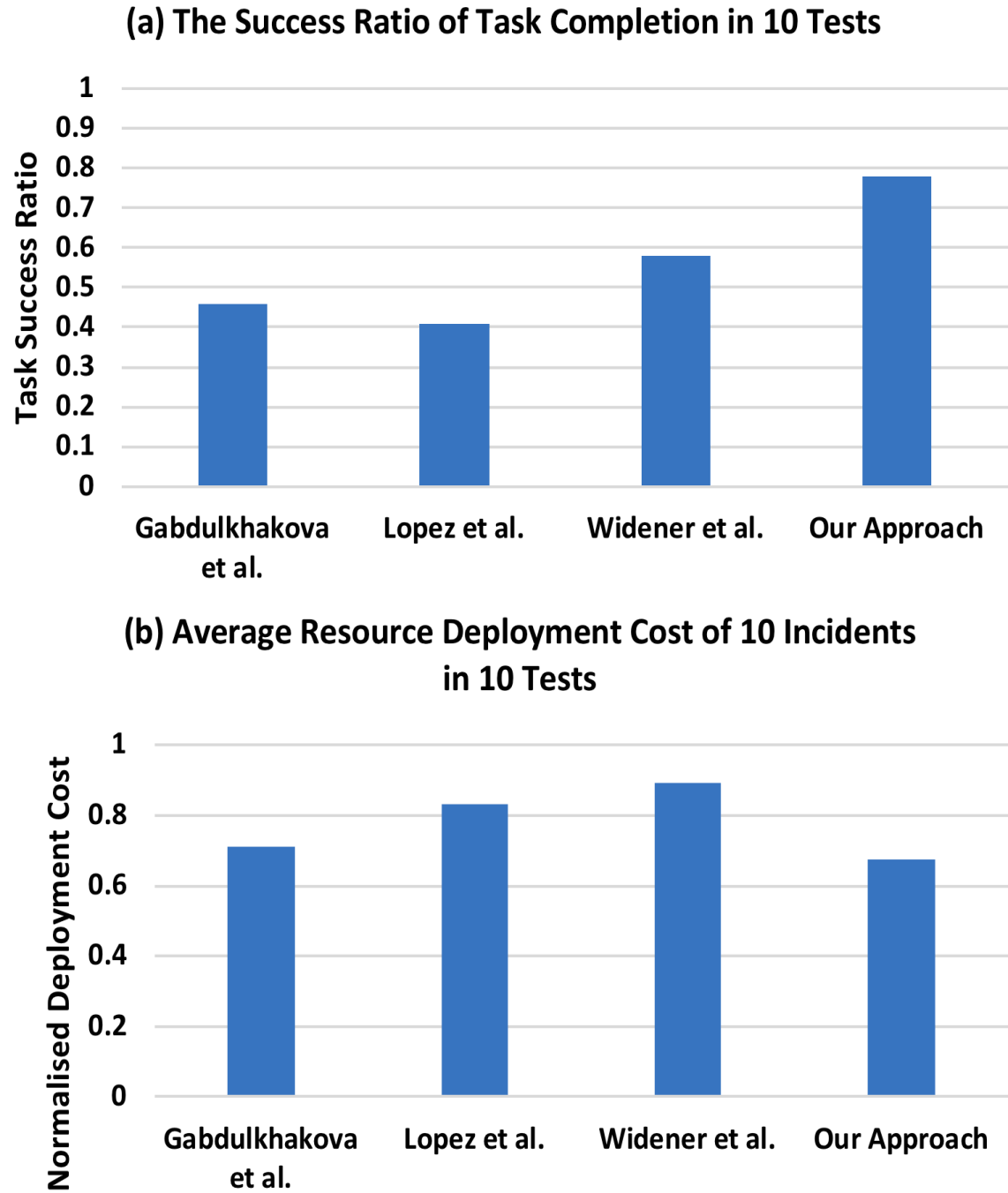


Figure 4.8: The Experimental Results of Scenario 3



We can see from Figure 4.6 (a), in Scenario 1, that our and Gabdulkhakova et al.'s approaches have a very high success ratio of the task completion (above 0.9), since both approaches have incorporated resource coordination mechanisms to address resource contention problems. Widener et al.'s approach is also capable of handling the problems of execution variation, but their approach only uses the distance factor as the optimisation objective, which results in the highest cost among all four approaches (see Figure 4.6 (b)). Our approach takes both resource money expenditure and deployment time into consideration, so we can ensure resources with appropriate attributes to be deployed to the most needed incidents.

From Figure 4.7 (a), we can see that after introducing both task and execution variations into Scenario 2, the task success ratios of all of four approaches have decreased in various degrees. Gabdulkhakova et al.'s and Widener et al.'s approaches have the most significant decreases (around 0.2), since their approaches fail to consider task variation. Our and López et al.'s approaches are capable of handling the problems that occur in task variations, such as releasing resources from cancelled tasks for the usage of incoming tasks, which results in a better control of the task failure comparing with the other two approaches in the same scenario.

In Figure 4.8 (a), we can see that the task success ratios of López et al.'s and Gabdulkhakova et al.'s approaches have decreased to around 0.4 in Scenario 3, because their approaches are incapable of handling incident variations. Our approach still has the highest task completion ratio (0.78) and the lowest resource deployment cost (0.67).

In summary, the results of the three experimental scenarios indicate that a resource deployment approach that fails to provide comprehensive mechanisms to handle all dynamic variations in emergency operations could cause the decrease of the task success rate and the increase of the resource deployment cost. The proposed approach in this chapter has the best performance in the experiments because of the capability of providing appropriate dynamic resource deployment mechanisms to address different problems caused by incident variation, task variation and execution

variation.

## 4.5 Summary

This chapter proposed an agent-based dynamic resource deployment approach to handle the emergency responses to multiple concurrent incidents, which provided solutions for achieving research **Objectives 3** and **4**, set up in Section 1.4.

The proposed approach took the consideration of dynamic changes in terms of incident variation, task variation and execution variation and provided three dynamic resource deployment mechanisms to address various dynamic problems during the resource deployment process. The experiment analysis proved that the proposed approach could not only efficiently minimise the total resource deployment cost of multiple incidents based on resource money expenditure and deployment time, but also ensure high success rates of rescue tasks.

## Chapter 5

# A Smart Emergency Response System in Real-World Environments

In this chapter, an agent-based smart emergency response system is proposed to automatically generate resources deployment plans to multiple emergency incidents, which provides a solution for solving research **Issue 5**, identified in Section 1.4. The proposed system has three major features, which are that **(1)** the proposed system is designed to automatically coordinate multiple departments to solve practical emergency response problems in the real-world environments; **(2)** the proposed system contains a high-level resource rationalisation model to optimise the resource usage for emergency departments with different response codes, which can be widely used and adapted in different regions and countries. and **(3)** the proposed system architecture is designed with high extendability, which can easily incorporate with different emergency response strategies, optimisation algorithms and dynamic mechanisms (i.e the three mechanisms, introduced in Chapter 4) in the future.

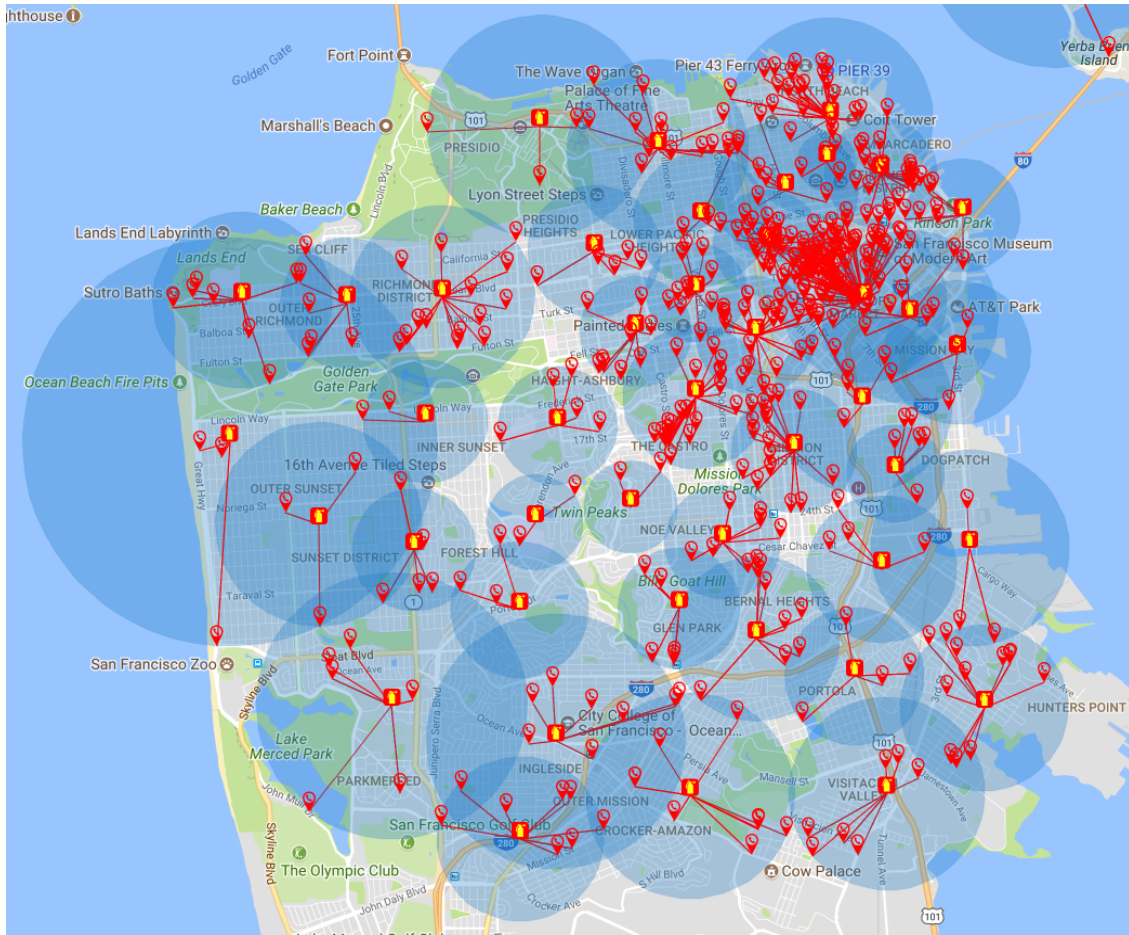
In the experiment, the proposed system is test along with emergency response data from San Francisco Fire Department and Police Department to evaluate the system performance for addressing real-life emergency incidents in urban areas.

The rest of this chapter is organised as follows. Section 5.1 gives detail of problem description. Section 5.2 describes a resource deployment model for resource usage

rationalisation and definitions. Section 5.3 presents the detail design of the proposed smart emergency response system. Section 5.4 explains the system deployment design and implementation in the real-world environments. Section 5.5 demonstrates the experiment settings and results. Section 5.6 summarises this chapter.

## 5.1 Problem Description

Small-scale emergency incidents are normally characterised by their unpredictability, uncertainty and variability. An emergency response for such an incident normally requires resource dispatchers to deploy and coordinate rescue resources from multiple emergency services within a short time to fulfil different requirements in an incident [BK16]. In real-world environments, most of emergency department around the world have already applied Computer-Aided Dispatch (CAD) systems to facilitate the resource deployment work for human resource dispatchers. However, due to the large amounts of emergency response demands in metropolitan regions and the complexity of the simultaneous resource deployment for multiple incidents, human resource dispatchers suffer tremendous pressures and difficulties for providing effective emergency responses even with the help of CAD systems.



**Figure 5.1:** San Francisco Fire Department Unit Dispatch Records for 500 Incidents

To further verify our concerns regarding the working efficiency of current emergency response systems, used by big cities around the world, this PhD study has accessed the open database of San Francisco Fire Department [18e] and visualised their unit dispatch records for emergency incidents on GoogleMap (demonstrated by Figure 5.1) as an example to demonstrate the flaws of real-world emergency response systems. Through some basic data analysis, two major problems are discovered in their emergency response system. First, due to the inefficiency of the manual unit dispatching process, the response time, (i.e. the time that from an emergency call received, to the time of relevant rescue units, being departed on route to the incident location), of an incident usually occupies 35% to 45% of the total unit deployment time. The human-base emergency response is time-consuming, especially for life-threatening incidents such as house fire. Second, from Figure 5.1, we can see that

each of the 500 incidents is only handled by a single fire station and the unit dispatchers tend to select response units from the nearest fire stations. From a practical point of view, this kind of response protocol is reasonable, since by avoiding the cooperation between fire stations in different districts and using distance as the most important factor to select resources for incidents can significantly simplify the resource management process. Nevertheless, from a long-term perspective, this kind of response protocol will increase the workloads of those fire stations, which have high incident rates in the surrounding areas (see the top right of Figure 1), thus to result in irrational resource usages.

Another problem of current emergency response systems in the real-world environments is the fixed resource optimisation objectives. In most real-life situations, emergency incidents in metropolitan regions usually come with different emergency response codes, representing the propriety or severity of incidents. Normally, incidents with different emergency response codes should have different objectives or constraints, which have to satisfy during the resource dispatching (deployment) process, such as resource money expenditure, resource deployment fairness, resource deployment time, task completing number, etc [HLH16; Su+14; WWS07; SL16; Yan+13]. However, due to the complexity of manually coordinating and dispatching different rescue resources from multiple emergency services, majority of emergency departments around the world can only consider the resource deployment time, (i.e. the time that from an incident call be received to the time that rescue can units arrive the incident location), as the most significant objective for the emergency response to an incident [BK16; LIB08; GB07]. Naturally, in San Francisco Fire Department, the distances between the fire stations and the incidents' locations are used as the sole factor to select fire units to conduct rescue operations, which results in inappropriate resource utilisation in most of the fire stations.

To overcome the limitations of current emergency response systems in the real-world environments, a smart emergency response system with a high-level resource rationalisation model is proposed in this chapter to handle the resource deployment

for incidents according to different emergency codes, cost attributes and objectives.

## 5.2 A Resource Deployment Model for Resource Usage Rationalisation and Definitions

To rationalise the usage of limited rescue resources for the emergency responses of small-scale incidents, a flexible resource deployment model is proposed in this section, which will be used in our smart emergency response system later (see Section 5.3 for detail) to generate resource deployment plans, based on the attributes of incidents and resources. The proposed model includes five important terms, which are *emergency response*, *rescue resource*, *resource deployment plan*, *cost function* and *objective function*.

More specifically, in the resource deployment model, an incident is described by an emergency response and a resource deployment plan contains a set of rescue resources that be selected to handle the emergency response based on the calculation of the pre-defined cost function and objective function. The detail definitions of the emergency response, rescue resource, resource deployment plan, cost function and objective function are described as below.

**Definition 17** (Emergency Response). An *emergency response*  $e$  is defined as a three-tuple,  $e = (con, cod, \mathbb{T})$ , where

- $con$  represents  $e$ 's content, such as medical emergency, vehicle accident, chemical leakage, etc.
- $cod$  represents  $e$ 's emergency response code.
- $\mathbb{T} = \{t_1, t_2, \dots, t_y\}$  represents a set of resource deployment tasks to be completed for  $e$ .

Furthermore, a resource deployment task  $t \in e.\mathbb{T}$  is further defined as a three-tuple,  $t = (dl, ser, \mathbb{R})$ , where

- $dl$  represents the resource deployment deadline.

- $ser$  represents the emergency service that  $t$  requires to make the response (i.e. police, medical and fire & rescue).
- $\mathbb{R} = \{r_1, r_2, \dots, r_m\}$  represents a set of required resources.

In the proposed resource deployment model, the values of emergency response codes and emergency services are not fixed, which can be defined and dynamically adjusted based on the needs of local emergency departments in different countries. Besides, it is assumed that local emergency departments are capable of estimating the resource deployment deadline of the tasks in an emergency response.

**Definition 18** (Rescue Resource). A *rescue resource*  $r$  is defined as a three-tuple,  $r = (typ, ser, \mathbb{C})$ , where

- $typ \in \{facility, mobile\}$  represents  $r$ 's type, where *facility* refers to unmovable resources such as hospitals, police stations, etc, while *mobile* refers to rescue vehicles and personnel.
- $ser$  represents the emergency services that  $r$  can provide.
- $\mathbb{C} = \{c_1, c_2, \dots, c_x\}$  represents a set of attributes that are related to the deployment cost of  $r$ , which could be resource quality, moving velocity, distance, scarcity, resource expenditure, etc.

**Definition 19** (Resource Deployment Plan). A *resource deployment plan*  $p$  for an emergency response is defined as a two-tuple,  $p = (e, \mathbb{R})$ , where  $\mathbb{R}$  represents a set of resources, proposed for emergency response  $e$ .

In summary, emergency incidents usually come with different emergency response codes to represent their priorities or severities. For most situations, a high priority emergency response might have different resource deployment objectives comparing with a low priority emergency response [Muk+17]. For example, the resource deployment for a high priority emergency response usually needs to choose resources with fast-moving velocities or short travel distances to minimise the resource deployment time, while a low priority emergency response might need to avoid using scarce



and expensive resources to reduce the money expenditure. To consider different objectives in different emergency response codes, in the proposed resource deployment model, a cost function with multiple cost attributes to select appropriate resources for emergency responses with different objectives, is defined by Definition 20 and calculated by Equation 5.1:

**Definition 20** (Cost Function). A *cost function* for a single resource's deployment to task  $e.t_y$  is defined by Equation 5.1:

$$CR(e.t_y, r) = \sum_{c_x \in e.C} (E_x(c_x) \times W_x(e.cod)), \quad (5.1)$$

where  $E_x$  is an evaluation function that converts the value of cost attribute  $c_x$  to a normalised value in-between 0 and 1, while  $W_x$  is a weighting function that determines the importance of cost attribute  $c_x$ .

Generally, each cost attribute  $c_x$  in cost function  $CR$  is related to a specific resource deployment objective. For example, a resource equipment's quality or scarcity is normally related to the money expenditure of the resource [WWS07]. Through adjusting the weighting of each cost attribute  $c_x$ , the cost (rationality) of deploying rescue resources to emergency incidents with different response codes can be calculated. The detail of the weighting function  $W_x$  depends on the structure of the emergency response codes that are used by local emergency departments in different countries.

Based on the above definitions, the objective function of the resource usage rationalisation for a set of emergency responses  $\mathbb{E} = \{e_1, e_2, \dots, e_n\}$  is defined by Equation 5.2:

$$OE = \arg \min_{\mathbb{P}^* = \{p_1^*, \dots, p_n^*\}} \sum_{e_n \in \mathbb{E}} CE(e_n, p_n^*) = \sum_{e_n.t_y \in e_n.T} \sum_{r_m \in p_{n_y}^*.R} CR(e.t_y, r_m) \quad (5.2)$$

subject to  $p_{n_y}^*.R \in p.R$ ,

where  $CE$  represents the cost function of the resource deployment for a single emergency response  $e$  and  $p^*$  represents the resource deployment plan with the minimal cost for emergency response  $e$ .

$CE$  indicates that the ultimate objective of the resource usage rationalisation is to search for an optimal resource deployment plan  $p_n^*$  for each of emergency response  $e_n \in \mathbb{E}$ , respectively, to minimise the total resource deployment cost, which can be calculated by using any appropriate multi-objective optimisation approach, such as the domain transportation theory, introduced in Section 3.2.

### 5.3 Design Detail of the Agent-Based Smart Emergency Response System for Multiple Incidents in Dynamic Environments

As described in Section 5.1, one of the major problems of current emergency response systems is their time-consuming processes in resource deployment, which heavily rely on human operators to manually send resource dispatching orders to different emergency departments through computer-aided dispatch systems. In this section, the design detail for a smart emergency response system is introduced to automatically coordinate multiple emergency departments or services to dispatch their resources for concurrent incidents in the real-world environments.

Overall, the whole smart emergency response system is designed as a decentralised agent-based multi-thread system, with each processing thread composed of an emergency response control component and a rescue resource management component to handle the emergency response of an incident.

The emergency response control component contains response agents, deployments agents and coordinator agents to address the emergency response requirements received from the outside world. Meanwhile, the rescue resource management component has facility agents and mobile agents distributed over metropolitan regions to conduct resource deployment plans. The agents used in the proposed system are based on the agents' definitions, introduced in Subsection 3.3.1 (i.e. response agents, deployments agents, facility agents and mobile agents) and Section 4.2 (i.e. coordinator agents).

**Design Consideration of Using a Decentralised System Architecture:**

There are two considerations of implementing the proposed smart emergency response system with a decentralised architecture instead of using a complete centralised design.

**Consideration 1: to overcome the resource deployment (dispatching) difficulty cross heterogeneous resource management systems.**

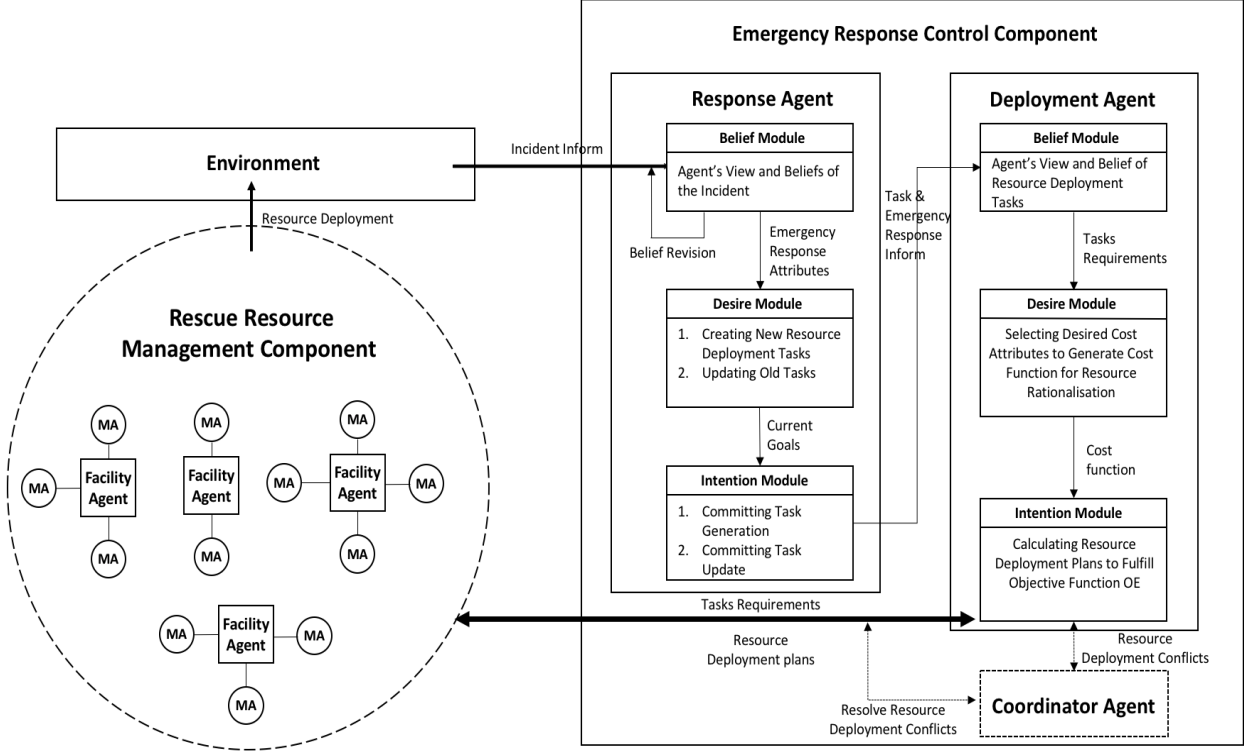
In the majority of countries, different emergency departments usually have their own resource management systems with different regulations to control their rescue resources [CPL14]. Implementing a centralised system to manage all resources that come from different emergency departments would be impractical to be realised in real-world environments. Besides, some emergency departments might have sensitive resource information that they do not want to share with other departments. In the proposed smart emergency response system, emergency facilities such as police stations and hospitals are controlled by decentralised facility agents, with each facility agent be implemented as customised resource management applications to manage relevant mobile agents in the facility.

**Consideration 2: to avoid the single point of fail, which refers to one system fault or malfunction causes the entire system to stop operating [Lyn09].**

For emergency response systems, there is no doubt that system availability and reliability are two of the most important factors need to consider during the system design. Although adding redundant backup systems can somehow avoid the single point of fail, but there is no guarantee that whether the backup systems will fail at the same time. Besides, redundant safety systems may result in more complexed system design, which more prone to errors [Sag04]. In the proposed smart emergency response system, rescue resources

are managed and controlled by independent facility agents, which mean the system fail of one facility agent will not affect the operations of other facility agents and their relevant mobile agents.

The architecture of the proposed smart emergency response system is depicted in Figure 5.2.



**Figure 5.2:** The Architecture of the Proposed Smart Emergency Response System

In detail, the Belief-Desire-Intention (BDI) agent model [R+95] is applied to implement the response and deployment agents in the emergency response control component [R+95] of an emergency response processing thread. In BDI agents, *beliefs* represent an agent's informational state of the world; *desires* refer to the agent's goals or objectives and *intentions* refer to the agent's commitments to its goals and its plans to achieve those goals. In BDI-based agent systems, an agent pursues its predefined goals (desires) by adopting appropriate plans (intentions) according to the current state of the agent's environment (beliefs). In the emergency resource control component of the proposed system, both a response agent and a deployment agent have three modules, which are a belief module, a desire module

and an intention module, respectively. BDI agent model has been widely used for the implementation of reasoning and planning functionalities in resource-bounded agents in dynamic environments, which is extremely suitable to handle small-scale incidents that are characterised by uncertainty and variability [MHL99].

The smart emergency response process for an incident involves one response agent, one deployment agent, multiple facility agents and mobile agents. A response process can be broken down into six major steps as follows.

**Step 1:** When the information of an incident be input into the system, a new processing thread with instances of a response agent  $ra$  and a deployment agent  $da$  will be generated to handle the emergency response  $e$  of this incident automatically. The received incident information will be first passed to response agent  $ra$  's belief module, in which emergency response content  $e.con$  and response code  $e.cod$  will be generated to represent response agent  $ra$  's beliefs of the incident, and they can continuously be updated through a belief revision function during the emergency response process.

**Step 2:** Next, emergency response content  $e.con$  and code  $e.cod$  are passed to response agent  $ra$  's desire module. The desire module contains two desires, which are (1) creating new resource deployment tasks and (2) updating old resource deployment tasks. At this point, response agent  $ra$  needs to select desires from the module as its current goals, which includes two scenarios.

**Scenario 1:** if emergency response content  $e.con$  and code  $e.cod$  come from a complete new incident,  $ra$  will choose desire (1) to create a set of new resource deployment task  $e.T$ . Each resource deployment task  $t \in e.T$  only requires resources from a particular emergency service (i.e.  $t.ser$ ).

**Scenario 2:** if emergency response content  $e.con$  and code  $e.cod$  come from  $ra$  's revised beliefs of an under-processing incident,  $ra$  will choose desire (2) to update the information of old tasks, which may includes task deadline  $t.dl$  and the detail of required resources  $t.R$ . In some situations,  $ra$  may also need

to choose desire (1) to add new resource deployment tasks to the emergency response's original task set  $e.\mathbb{T}$ . For example, a house fire incident originally has one task that requires several fire engines. But after the fire widely spreads and injures people, a new resource deployment task needs to be added to require ambulances and the old task might need to be updated to require more fire engines.

**Step 3:** After the desire selection process, response agent  $ra$  will commit its current goals in intention module, in which a pre-defined plan library with hierarchical decision trees can be used to atomically generate and update resource deployment tasks [Ven+08]. Finally, resource deployment tasks  $e.\mathbb{T}$  along with emergency response  $e$  will be sent to deployment agent  $da$  as its beliefs of the emergency response requirements.

**Step 4:** For each received resource deployment task  $t \in e.\mathbb{T}$ , deployment agent  $da$  will generate an internal thread to process its resource deployment requirements  $t.\mathbb{R}$ . In each task processing thread, a task's requirements  $t.\mathbb{R}$  will be passed from the belief module to the desire module, in which deployment agent  $da$  needs to choose a set of desired resource cost attributes  $r.\mathbb{C}$  (see Definition 18) to compose a cost function  $CR$  (i.e. Equation 5.1) for resource usage rationalisation. To automatically generate appropriate cost function for a task, emergency experts might need to set up inference rules or engines in deployment agent  $da$ 's desire module to infer cost attributes and their associated weighting based on the emergency response codes and protocols used by local emergency departments.

**Step 5:** After an cost function  $CR$  be generated for a task  $t_y$ , deployment agent  $da$  can start to calculate a resource deployment plan with the minimal cost for task  $t_y$ . First, deployment agent  $da$  will distribute the information of task  $t_y$  to nearby facility agents that provide the same emergency service as  $t_y.ser$ . For example, a task  $t$  that requires medical service (i.e.  $t.ser = medical$ ) will be distributed to hospitals. The distribution radius can be estimated by the task deadline  $t_y.dl$ . A facility agent

$fa_i$  who received the task information will propose an resource deployment plan  $p_i$  based on resources that under its control (i.e.  $fa.\mathbb{R}$ ), and plan  $p_i$  will be added to deployment agent  $da$  's plan library  $\mathbb{P}_y$ . After collecting resource deployment plans from nearby facility agents, deployment agent  $da$  will calculate an optimal resource deployment plan  $p_y^*$  based on all available resources in the plan library  $\mathbb{P}_y$  to fulfil objective function OE (see Equation 5.2 for detail). Finally, deployment agent  $da$  will send out plan  $p_y^*$  to relevant facility agents for the resource deployment of task  $t_y$ .

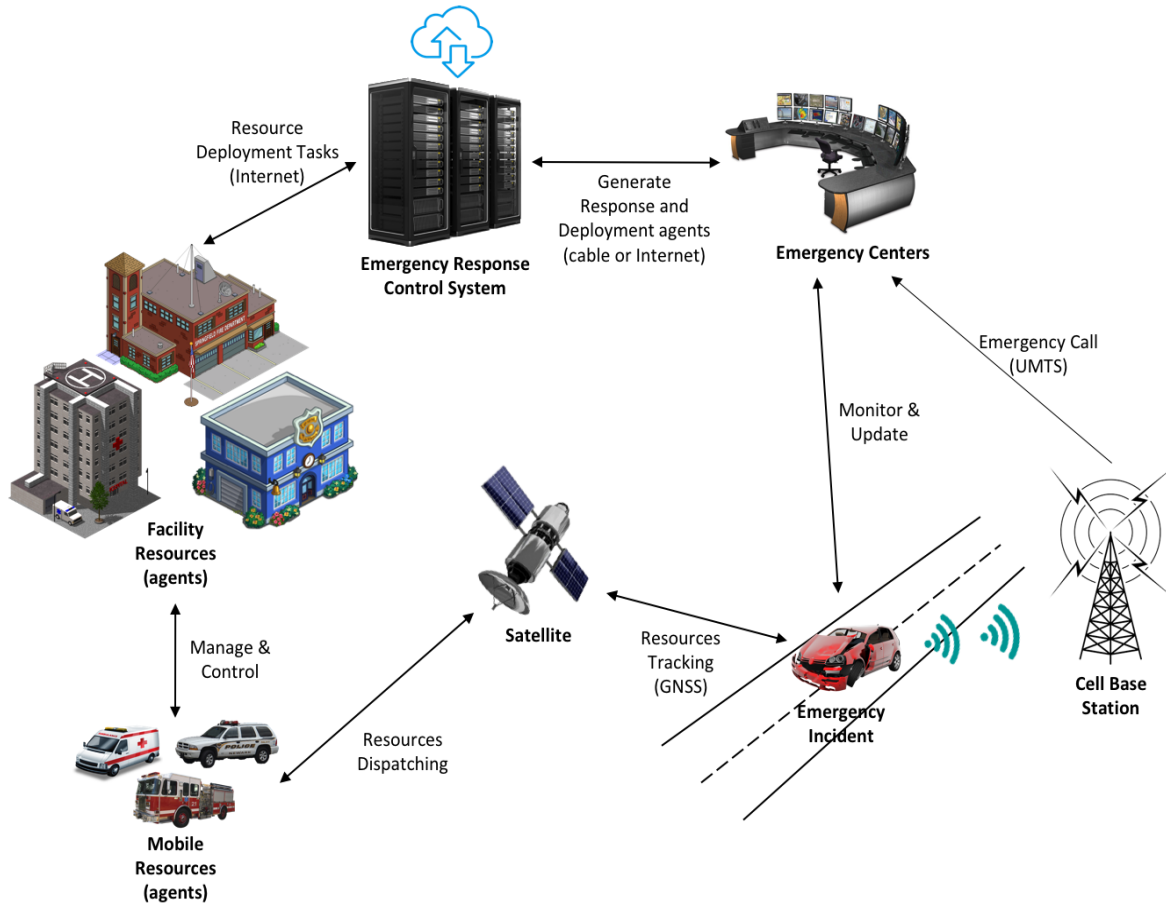
**Step 6:** After relevant facility agents receive the resource deployment  $p_y^*$  and send out required mobile resources to conduct the rescue operation for emergency response  $e$ , the response agent  $ra$  will continue monitor the the emergency response process and update its belief model for dynamic changes caused by different variations until the completion of emergency response  $e$ .

In addition to the above steps in an emergency response process, a coordinator agent  $ca$  might be dynamically generated during the resource deployment process of an emergency response to address resource deployment conflicts, such as resource contention problems between multiple emergency responses.

## 5.4 System Deployment in Real-World Environments

In real-world environments, the proposed smart emergency response system will be operated by one emergency centre or multiple emergency centres (see Figure 5.3).

More precisely, the proposed system can be installed on a local data centre and connected to emergency centres through cables, or a more flexible solution could be deploying the system on cloud servers and communicating with emergency centres through the Internet. The main responsibility of emergency centre is to pick up emergency calls from people and input incidents information into the smart emergency response system, then the response and deployment agents in the emer-



**Figure 5.3:** The Deployment of Smart Emergency Response System

gency response control component will be automatically generated to deploy and coordinate resources to incidents as described in the previous section. In addition to the emergency calls, the next generation emergency detection and analysis applications such as eCall [18d], AIDER [18a] and E-health care [Pau+09] can also be used to automatically report and update incident information into the smart emergency response control system through Universal Mobile Telecommunications System (UMTS) [PT03].

For the deployment of the decentralised rescue resource management component, facility agents will be installed on the servers of local emergency facilities as programs, which enable the communication between the local resource management systems and the deployment agents in the emergency response control component through the Internet. Mobile agents will mostly be deployed on rescue personnel and vehicles' portable devices as mobile applications, which can be tracked and



controlled through Global Navigation Satellite System (GNSS) [GFE12].

## 5.5 Experiments

### 5.5.1 Experimental Settings

An experiment was conducted to test the proposed smart emergency response system in real-world environments by using the emergency response requirements data from San Francisco Fire Department [18e] and Police Department [18g]. In the experiment, an agent-based emergency response simulator was developed based on the system architecture introduced in Section 5.3 (i.e. see Figure 5.4 in the next page). We used Java to implement the multithreading emergency response system and used Jade [BPR99] as the basic framework to design the response agents, deployment agents, facility agents, mobile agents and coordinator agents. Furthermore, Jadex engine [BLP03] was used to implement the BDI-infrastructure for the response and deployment agents.

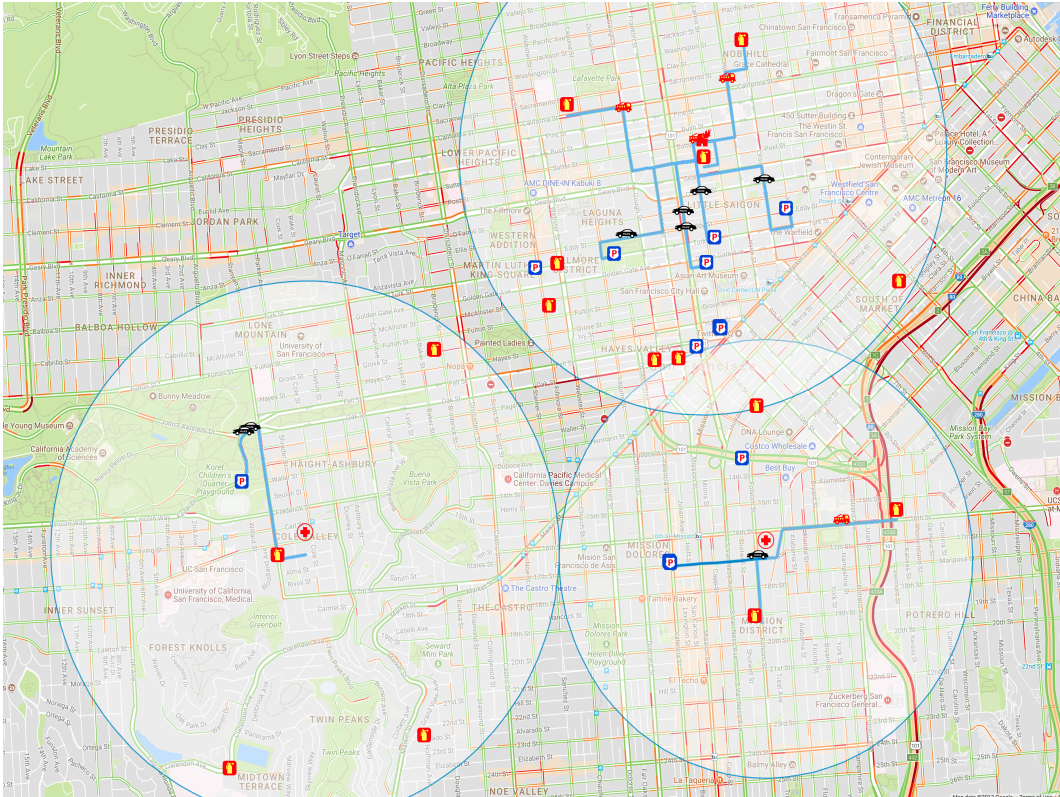


Figure 5.4: Smart Emergency Response Simulator

The main purpose of the experiment is to evaluate the performance of the proposed smart emergency response system regarding resource usage rationalisation for incidents with different emergency response codes and compare the resource deployment results with real-world emergency response data. In order to do so, the open databases of San Francisco Fire Department and Police Department were accessed and 29233 incidents that involved the unit dispatching from both the departments during 2016 were extracted from the databases. The open databases contain detail information in terms of incidents (i.e. incident content, response code, etc.), rescue units that be dispatched to the incidents (i.e. unit type, unit service, etc.) and unit response and deployment times.

During the experiment, our emergency response simulator was used to deploy resources for the 29233 test incidents. Two of the most commonly used cost attributes in emergency response were considered to calculate the resource deployment cost, which were resource deployment time and resource money expenditure. A resource's deployment time to an incident can be estimated by using the resource's average moving velocity and the resource distance to the incident, while a resource's money expenditure is mostly based on the information provided by the websites of San Francisco Fire Department [18h] and Police Department [18i]. Furthermore, San Francisco emergency departments use the priority-based emergency response codes (i.e. 1, 2 and 3) to classify incidents, with 3 stands for the highest priority and 1 stands for the lowest priority. Based on the above information, the cost function used in the experiment is defined by Equation 5.3:

$$CR'(e.t_y, r) = (E_t(\frac{r.dis}{r.vel})W_t(e.cod) + E_m(r.exp)W_m(e.cod)) \times DL(e.t_y, r), \quad (5.3)$$

where *dis* represents *r*'s shortest path to emergency response *e*; *vel* represents *r*'s average moving velocity and *exp* represents *r*'s money expenditure.  $E_t$  and  $E_m$  are two evaluation functions to convert the value of a resource's deployment time and money expenditure to a normalised value in-between 0 and 1.  $W_t$  and  $W_m$  are

functions used to calculate the weightings of resource deployment time and money expenditure, respectively, which are defined by the following equation.

$$\begin{cases} W_t(e.cod) = \frac{e.cod}{3} \\ W_m(e.cod) = 1 - \frac{e.cod}{3}. \end{cases} \quad (5.4)$$

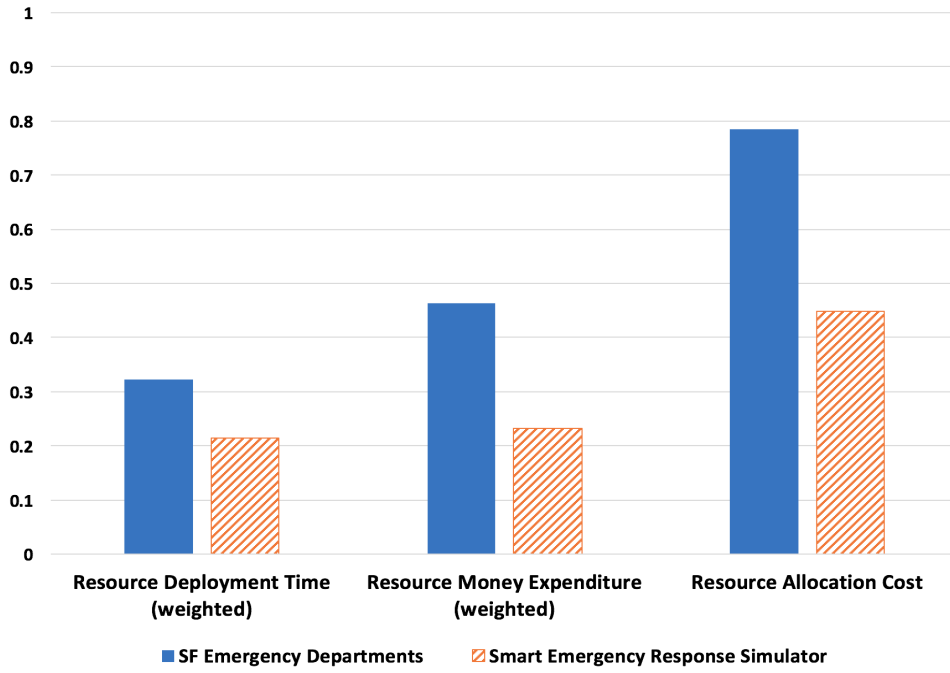
Besides,  $DL$  is a function, used to determine whether  $r$  can be delivered within task  $e.t_y$ 's deadline, which is further defined by Equation 5.5:

$$DL(e.t_y, r) = \begin{cases} 1, & \text{if } \frac{r.dis}{r.vel} \leq e.t_y.dl \\ +\infty, & \text{if } \frac{r.dis}{r.vel} > e.t_y.dl. \end{cases} \quad (5.5)$$

The cost function  $CR'$  i.e. Equation 5.3 was also used to calculate the emergency response costs for the 29233 test-incidents based on the real unit dispatching records stored in the databases of San Francisco Fire Department and Police Department, and the results were recorded for the comparison analysis with the deployment costs generated by our emergency response simulator.

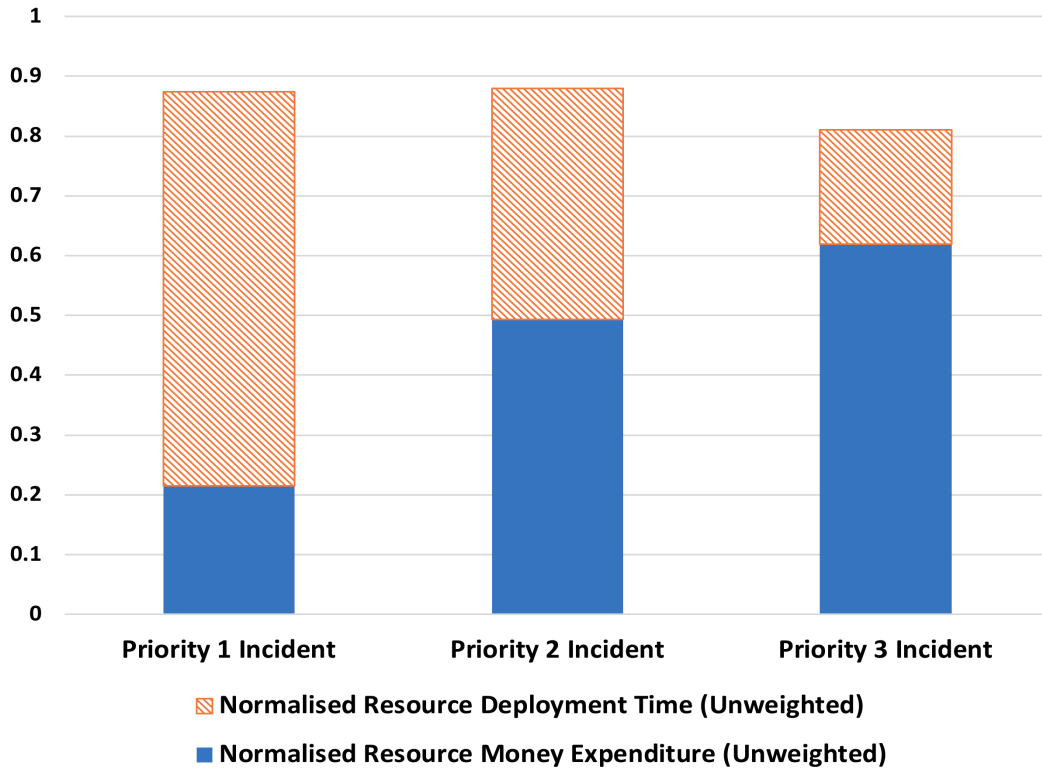
### 5.5.2 Experimental Results and Analysis

Figure 5.5 shows the average resource deployment time, money expenditure and cost for an incident in 29233 test cases based on the resource deployment of the proposed system and the real emergency response data from San Francisco emergency departments. All the calculation results were normalised in-between 0 and 1 for better comparison and analysis.



**Figure 5.5:** Average Resource Deployment Time, Money Expenditure and Cost for an Incident in 29233 Test Cases

From Figure 5.5, we can see that San Francisco emergency departments need average 0.785 normalised resource deployment cost to process the emergency response of an incident in their normal operation. A big portion of the cost comes from the resource money expenditure, which is due to San Francisco emergency departments only consider the distance factor during the resource deployment process. However, the proposed system in this chapter only requires average 0.448 resource deployment cost to process an incident. This is because the smart emergency response system can automatically select and deployment resources from multiple emergency departments, which completely eliminates the resource dispatching time, required by traditional emergency response systems. Besides, the proposed resource deployment model takes both resource deployment time and money expenditure into the consideration and generate appropriate resource deployment plans based on incidents' response codes. Naturally, the proposed system outperformed San Francisco emergency departments in both resource deployment time (0.215 verse 0.322) and money expenditure (0.233 verse 0.463). The comparison results convince the advantages of the proposed system.



**Figure 5.6:** The Resource Usage Rationalisation Based On Incident Priority

Figure 5.6 further demonstrates that the proposed system can effectively rationalise resource usage for incidents with different priorities. For example, for incidents with priority 3 response code, the resource deployment time only occupies 19% percent of the total resource deployment cost, which means the proposed system has selected resources with low deployment times to make fast responses to these high priority incidents. For incidents with priority 1 response code, the proposed system has selected resources with low money expenditure (21% of the total cost) to help emergency departments to control their budgets.

## 5.6 Summary

In this chapter, an agent-based smart emergency response system was proposed to automatically handle the emergency response for multiple incidents in the real-world environments based on different emergency response codes, cost attributes and objectives, provided a solution for achieving research **Objective 5**, set up in Section 1.4.

The proposed system incorporated a flexible resource deployment model to rationalise resource usage for multiple emergency governments in different regions and countries. Furthermore, the architecture of the propose system was composed of a emergency response control component and a resource management component, in which BDI agent model was used for agent designs, which can constantly update dynamic changes caused by different variations. The proposed system architecture was also designed with high extendability that can easily utilise with other mechanisms to enhance the power of the system in different situations, such as employing the three dynamic resource deployment mechanisms, introduced in Section 4.3, to handle different variations.

The proposed system was evaluated by the real-world data from San Fransisco Fire Department and Police Department, and its performance was also compared with the traditional emergency response process from San Fransisco emergency departments. The evaluation results indicated that the proposed system could generate efficient resource deployment plans to rationalise resource usage for real-world emergency incidents, and outperformed the response systems used by San Fransisco emergency departments, in terms of both resource deployment time and money expenditure.

# Chapter 6

## Conclusion and Future Work

In metropolitan regions, emergency responses for incidents with different severity levels, usually require multiple resources that have appropriate functionalities, money expenditure, moving velocities to conduct a rescue process. These resources could be distributed over an extensive area with different ownerships. Solving the resource deployment problem for the emergency responses of these incidents involves complicated collaboration between multiple emergency departments under strict time constraints. Traditional resource deployment approaches usually have difficulties to efficiently find out the best resource assignment within the time limits by considering the large number of possibilities, which result in a considerable increase in fatalities and financial loss. The aim of this thesis is to research agent-based resource deployment strategies to improve the emergency responses for the incidents that happen around urban areas. During this PhD study, several resource deployment models, algorithms and mechanisms were proposed, and an agent-based framework and an agent-based smart emergency response system were developed to handle different resource deployment problems during emergency response processes. This chapter summarises the major contributions of this thesis and outlines the potential future work.

## 6.1 Contributions of This Thesis

The contributions of this thesis are summarised as follows.

- 1. An innovation approach for optimal resource deployment in emergency management was proposed.**

The proposed approach introduced a comprehensive resource deployment model to represent the complicated relationships and attributes among rescue resources, emergency incidents, resource deployment tasks, plans, cost functions and objective functions. Furthermore, the proposed approach presented an agent-based framework to effectively deployment rescue resources to a single emergency incident in a decentralised manner. More precisely, this system used a response agent to generate resource deployment tasks for an incident and employed a deployment agent to simultaneously calculate resource deployment solutions for these tasks, in which the domain transportation theory was applied to effectively generate the optimal plan by converting the resource deployment problem to a resource mapping problem. Comparing with centralised approaches, the proposed approach was more applicable and practical to be used to handle emergency incidents involving the collaboration of multiple emergency departments and private resource companies. Besides, the simultaneous task handling mechanism in the proposed approach provided more efficient resource deployment speed comparing with the speed of sequential task deployment approaches. The experimental results demonstrated the good performance of the proposed approach in terms of both the resource deployment cost and the computational cost (i.e. speed) for generating the optimal resource deployment plans.

- 2. An agent-based dynamic resource deployment approach for improving emergency responses to concurrent incidents was proposed**

The proposed approach was designed to efficiently coordinate multiple emergency departments with different types of agents to rationalise the usage of



their limited rescue resources for the emergency responses of concurrent incidents and to automatically adjust resource deployment plans through resource coordination, reassignment and release to achieve the adaption against the variations that can occur during the emergency rescue operations. To achieve the adaption abilities, the proposed approach introduced three adaptive resource deployment mechanisms for addressing the dynamic problems caused by incident variation, task variation and execution variation during the resource deployment process. In addition, deployment agents were dynamically generated in the proposed approach to handle resource contention problems among multiple emergency incidents by effectively finding replacement resources and reassigning these resources to appropriate incidents. The experiment results convinced that the proposed dynamic mechanisms could significantly increase the success rate of the resource deployment for concurrent incidents.

### **3. A deployable smart emergency response system for real-world emergency situations was developed.**

The proposed system was designed with high practicability and extendability to be easily deployed in the real-world environments. The system could automatically provide smart emergency responses for the resource deployment of concurrent emergency incidents based on multiple objectives. In the proposed system, a flexible resource deployment model was employed to deploy rescue resources for emergency departments in different regions and countries. The architecture of the system was composed of an emergency response control component and a resource management component, in which BDI agent model was used to implement response agents and deployment agents. The BDI agent model provided the system with high adaptability and extendability to employ different optimisation algorithms, resource deployment strategies and dynamic mechanisms, which could further improve the system's ability for solve real-life emergency resource deployment problems. In the experiment, the proposed system was evaluated by using the emergency response data from

San Francisco emergency departments. The evaluation results demonstrated that the proposed system could not only help public emergency departments to effectively reduce the average time of response deployment for the real-life emergency response requirements, but also rationalised the usage of expensive resources for appropriate incidents.

## 6.2 Future Work

Although the resource deployment approaches and systems proposed in this thesis could provide promising solutions for solving different resource deployment problems for the emergency responses of incidents, there still exists some space to improve for these approaches and systems. The future work includes the two major directions.

### **1. Combinations of data mining and machine learning methods to enhance the performance of the approaches, proposed in Chapters 3 and 4**

The proposed resource deployment approaches in Chapters 3 and 4 relayed on appropriate cost functions to generate the optimal resource deployment plans for emergency incidents, in which resource deployment time and money expenditures were set up as the optimisation objectives for different types of incidents. In real-life situations, some special types of incidents might not have predefined or well-defined cost functions to be used in the proposed approaches. This could decrease the resource deployment efficiency. Therefore, one of the future directions after this PhD study is to integrate machine learning and data mining methods into the proposed approaches to evaluate the efficiency of emergency responses and analysis the resource deployment results for different types of incidents. By doing so, the proposed approaches should be able to discover the correlations between cost factors and a specific type of incidents, so as to improve the designs for the cost functions in this type of incidents.

**2. Extension of the smart emergency response system, proposed in Chapter 5 by utilising the three dynamic mechanisms, introduced in Chapter 4**

The proposed smart emergency response system in Chapters 5 was designed to be able to integrate different resource deployment algorithms and mechanisms to enhance the system's abilities. Therefore, another future direction is to extend the system by integrating the three dynamic resource deployment mechanisms, introduced in Chapter 4 into the BDI agents' desire and intention models in the smart emergency response system. By doing so, the ability for the system to handle dynamic problems caused by incident variation, task variation and execution variation for the real-life emergency situations, would be significantly improved.

# Bibliography

- [18a] *Accident information and driver emergency rescue [Online]*. 2018. URL: <http://www.transport-research.info/project/accident-information-and-driver-emergency-rescue>.
- [18b] *Basic & Advanced Life Support Ambulance*. 2018. URL: <https://www.maas911.com/services/basic-and-advanced-life-support-ambulance/>.
- [18c] *Emergency Service Response Codes*. 2018. URL: <https://en.wikipedia.org/wiki/Emergency-service-response-codes>.
- [18d] *European Commission eSafety Initiative [Online]*. 2018. URL: <http://ec.europa.eu/information%20society/activities/esafety/indexen.htm>.
- [18e] *Fire Department Calls for Service*. 2018. URL: <https://data.sfgov.org/Public-Safety/Fire-Department-Calls-for-Service/nuek-vuh3>.
- [18f] *OnStar by GM*. 2018. URL: <http://www.onstar.com/>.
- [18g] *Police Department Calls for Service*. 2018. URL: <https://data.sfgov.org/Public-Safety/Police-Department-Calls-for-Service/hz9m-tj6z>.
- [18h] *San Francisco Fire Department*. 2018. URL: <https://data.sfgov.org/Public-Safety/Police-Department-Calls-for-Service/hz9m-tj6z>.
- [18i] *San Francisco Police Department*. 2018. URL: <https://data.sfgov.org/Public-Safety/Police-Department-Calls-for-Service/hz9m-tj6z>.
- [18j] *The Australian Emergency Management*, <http://www.em.gov.au>. 2018. URL: <http://www.em.gov.au>.

- [Abo+15] Patrick Aboagye-Sarfo et al. “Growth in Western Australian emergency department demand during 2007–2013 is due to people with urgent and complex care needs”. In: *Emergency Medicine Australasia* 27.3 (2015), pp. 202–209.
- [Abo+16] Patrick Aboagye-Sarfo et al. “Impact of population ageing on growing demand for emergency transportation to emergency departments in Western Australia, 2005–2020”. In: *Emergency Medicine Australasia* 28.5 (2016), pp. 551–557.
- [Ada+08] NM Adams et al. “The aladdin project: intelligent agents for disaster management”. In: *International Workshop on Robotics for Risky Interventions and Surveillance of the Environment*. International Advanced Robotics Programme, 2008.
- [AH01] Selim Aksoy and Robert M Haralick. “Feature normalization and likelihood-based similarity measures for image retrieval”. In: *Pattern recognition letters* 22.5 (2001), pp. 563–582.
- [Ala+14] Zubaida Alazawi et al. “A smart disaster management system for future cities”. In: *Proceedings of the 2014 ACM international workshop on Wireless and mobile technologies for smart cities*. ACM, 2014, pp. 1–10.
- [Alt12] Nezih Altay. “Capability-based resource allocation for effective disaster response”. In: *IMA Journal of Management Mathematics* 24.2 (2012), pp. 253–266.
- [Bal+06] Vidhya Balasubramanian et al. *DrillSim: a simulation framework for emergency response drills*. Springer Berlin Heidelberg, 2006, pp. 237–248.
- [BCG07] Fabio Luigi Bellifemine, Giovanni Caire, and Dominic Greenwood. *Developing multi-agent systems with JADE*. Vol. 7. John Wiley & Sons, 2007.
- [Ber+05] Dave Berry et al. “FireGrid: Integrated emergency response and fire safety engineering for the future built environment”. In: UK e-Science Programme All Hands Meeting. 2005.
- [Bil+14] Holger Billhardt et al. “Dynamic coordination of ambulances for emergency medical assistance services”. In: *Knowledge-Based Systems* 70 (2014), pp. 268–280.

- [BJL06] John F Buford, Gabriel Jakobson, and Lundy Lewis. “Multi-agent situation management for supporting large-scale disaster relief operations”. In: *International Journal of Intelligent Control and Systems* 11.4 (2006), pp. 284–295.
- [BK16] A Bozorgi-Amiri and M Khorsi. “A dynamic multi-objective location–routing model for relief logistic planning under uncertainty on demand, travel time, and cost parameters”. In: *The International Journal of Advanced Manufacturing Technology* 85.5-8 (2016), pp. 1633–1648.
- [BLP03] Lars Braubach, Winfried Lamersdorf, and Alexander Pokahr. *Jadex: Implementing a BDI-infrastructure for JADE agents*. Citeseer, 2003.
- [BMM11] Swati Basak, Neelam Modanwal, and Bireshwar Dass Mazumdar. “Multi-agent based disaster management system: a review”. In: *International Journal of Computer Science & Technology* 2.2 (2011), pp. 343–348.
- [BP12] Federico Bergenti and Agostino Poggi. “Developing smart emergency applications with multi-agent systems”. In: *Emerging Communication Technologies for E-Health and Medicine*. IGI Global, 2012, pp. 31–44.
- [BPR99] Fabio Bellifemine, Agostino Poggi, and Giovanni Rimassa. “JADE–A FIPA-compliant agent framework”. In: *Proceedings of the Practical Applications of Intelligent Agents*. 1999, pp. 33–45.
- [Bre89] Friedrich Breyer. “On the intergenerational Pareto efficiency of pay-as-you-go financed pension systems”. In: *Journal of Institutional and Theoretical Economics (JITE)/Zeitschrift für die gesamte Staatswissenschaft* 145.4 (1989), pp. 643–658.
- [CH00] Myong-Hun Chang and Joseph E Harrington. “Centralization vs. decentralization in a multi-unit organization: A computational model of a retail chain as a multi-agent adaptive system”. In: *Management Science* 46.11 (2000), pp. 1427–1440.
- [Che+05] Rui Chen et al. “Design principles of coordinated multi-incident emergency response systems”. In: *International Conference on Intelligence and Security Informatics*. Springer. 2005, pp. 81–98.
- [Che+08] Rui Chen et al. “Coordination in emergency response management”. In: *Communications of the ACM* 51.5 (2008), pp. 66–73.
- [Chi+08] Pavan Kumar Chitumalla et al. “Emergency response applications: Dynamic plume modeling and real-time routing”. In: *IEEE Internet Computing* 12.1 (2008), pp. 38–44.

- [Cho+14] Jui-Sheng Chou et al. “Biological-based genetic algorithms for optimized disaster response resource allocation”. In: *Computers & Industrial Engineering* 74 (2014), pp. 52–67.
- [Cla13] Jeffrey J Clawson. *Method and system for the fire response dispatch protocol of an emergency dispatch system*. US Patent 8,417,533. Apr. 2013.
- [CPL14] Eileen J Carter, Stephanie M Pouch, and Elaine L Larson. “The relationship between emergency department crowding and patient outcomes: a systematic review”. In: *Journal of Nursing Scholarship* 46.2 (2014), pp. 106–115.
- [CTC07] Mei-Shiang Chang, Ya-Ling Tseng, and Jing-Wen Chen. “A scenario planning approach for the flood emergency logistics preparation problem under uncertainty”. In: *Transportation Research Part E: Logistics and Transportation Review* 43.6 (2007), pp. 737–754.
- [DH91] Thomas E Drabek and Gerard J Hoetmer. *Emergency management: Principles and practice for local government*. International city management association Washington, DC, 1991.
- [DPW11] Richard J Dawson, Roger Peppe, and Miao Wang. “An agent-based model for risk-based flood incident management”. In: *Natural Hazards* 59.1 (2011), pp. 167–189.
- [ED00] Kadir Ertogral and S David Wu. “Auction-theoretic coordination of production planning in the supply chain”. In: *IIE transactions* 32.10 (2000), pp. 931–940.
- [Fer99] Jacques Ferber. *Multi-agent systems: an introduction to distributed artificial intelligence*. Addison-Wesley Reading, 1999.
- [Fie06] Frank Fiedrich. “An HLA-based multiagent system for optimized resource allocation after strong earthquakes”. In: *Proceedings of the 38th Conference on Winter simulation*. 2006, pp. 486–492.
- [GB07] Qiang Gong and Rajan Batta. “Allocation and reallocation of ambulances to casualty clusters in a disaster relief operation”. In: *IIE Transactions* 39.1 (2007), pp. 27–39.
- [GF06] Veysel Gazi and Barış Fidan. “Coordination and control of multi-agent dynamic systems: Models and approaches”. In: *International Workshop on Swarm Robotics*. Springer. 2006, pp. 71–102.
- [GFE12] Peter Gaal, Sven Fischer, and Stephen W Edge. *Global navigation satellite system*. US Patent 8,334,807. Dec. 2012.

- [GKR11] Aygul Gabdulkhakova, B Konig-Ries, and Dmitry A Rizvanov. “An agent-based solution to the resource allocation problem in emergency situations”. In: *2011 Ninth IEEE European Conference on Web Services (ECOWS)*. 2011, pp. 151–157.
- [Gyö+09] Cornelia Györödi et al. “Web 2.0 Technologies with jQuery and Ajax”. In: *Journal of Computer Science & Control Systems* 2.2 (2009), pp. 11–15.
- [Haw+12] Glenn I Hawe et al. “Agent-based simulation for large-scale emergency response: A survey of usage and implementation”. In: *ACM Computing Surveys (CSUR)* 45.1 (2012), pp. 8–59.
- [Haw+15] Glenn I Hawe et al. “Agent-based simulation of emergency response to plan the allocation of resources for a hypothetical two-site major incident”. In: *Engineering Applications of Artificial Intelligence, Part B* 46 (2015), pp. 336–345.
- [HBC13] George Haddow, Jane Bullock, and Damon P Coppola. *Introduction to emergency management*. Butterworth-Heinemann, 2013.
- [HLH16] CL Hu, X Liu, and YK Hua. “A bi-objective robust model for emergency resource allocation under uncertainty”. In: *International Journal of Production Research* 54.24 (2016), pp. 7421–7438.
- [JAT13] Manish Jain, Bo An, and Milind Tambe. “Security games applied to real-world: Research contributions and challenges”. In: *Moving Target Defense II*. Springer, 2013, pp. 15–39.
- [Joh15] Athena N Johnson. “Emergency Relief Planning and Management Through Dynamic Agent-based Reallocation and Tasking (DART)”. In: *Procedia Engineering* 107 (2015), pp. 168–171.
- [Kap08] Naim Kapucu. “Collaborative emergency management: better community organising, better public preparedness and response”. In: *Disasters* 32.2 (2008), pp. 239–262.
- [Kie+09] Christopher Kiekintveld et al. “Computing optimal randomized resource allocations for massive security games”. In: *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems*. 2009, pp. 689–696.
- [Kit+99] Hiroaki Kitano et al. “Robocup rescue: Search and rescue in large-scale disasters as a domain for autonomous agents research”. In: *1999 IEEE International Conference on Systems, Man, and Cybernetics*. Vol. 6. IEEE. 1999, pp. 739–743.



- [KKT10] Sven Koenig, Pinar Keskinocak, and Craig A Tovey. “Progress on Agent Coordination with Cooperative Auctions.” In: *Proceedings of the 24th AAAI Conference on Artificial Intelligence*. 2010, pp. 1713–1717.
- [Kol+15] Panayiotis Kolios et al. “A resource-based decision support tool for emergency response management”. In: *Information and Communication Technologies for Disaster Management (ICT-DM)*. 2015, pp. 159–165.
- [Lam+15] Sean Shao Wei Lam et al. “Dynamic ambulance reallocation for the reduction of ambulance response times using system status management”. In: *The American Journal of Emergency Medicine* 33.2 (2015), pp. 159–166.
- [LIB08] Beatriz López, Bianca Innocenti, and Dídac Busquets. “A multiagent system for coordinating ambulances for emergency medical services”. In: *IEEE Intelligent Systems* 23.5 (2008), pp. 50–57.
- [Liu11] Jiakun Liu. “REGULARITY OF MONGE–AMPÈRE EQUATIONS IN OPTIMAL TRANSPORTATION”. In: *Bulletin of the Australian Mathematical Society* 83.01 (2011), pp. 173–176.
- [LL13] Ming Liu and Jiang Liang. “Dynamic optimization model for allocating medical resources in epidemic controlling”. In: *Journal of Industrial Engineering and Management* 6.1 (2013), p. 73.
- [LP00] Jerome H Lemelson and Robert D Pedersen. *Danger warning and emergency response system and method*. US Patent 6,084,510. July 2000.
- [LR01] Avraham Leff and James T Rayfield. “Web-application development using the model/view/controller design pattern”. In: *Proceedings of the 5th IEEE International Conference on Enterprise Distributed Object Computing*. 2001, pp. 118–127.
- [LT08] Lefei Li and Shuming Tang. “An artificial emergency-logistics-planning system for severe disasters”. In: *IEEE Intelligent Systems* 23.4 (2008), pp. 86–88.
- [Lyn09] Gary S Lynch. *Single point of failure: The 10 essential laws of supply chain risk management*. John Wiley and Sons, 2009.
- [Mar+09] Ramon Martí et al. “Providing early resource allocation during emergencies: The mobile triage tag”. In: *Journal of Network and Computer Applications* 32.6 (2009), pp. 1167–1182.

- [Mar+10] Francisco J Martinez et al. “Emergency services in future intelligent transportation systems based on vehicular communication networks”. In: *IEEE Intelligent Transportation Systems Magazine* 2.2 (2010), pp. 6–20.
- [MBT13] Xin Miao, David Banister, and Yanhong Tang. “Embedding resilience in emergency resource management to cope with natural hazards”. In: *Natural hazards* 69.3 (2013), pp. 1389–1404.
- [Men+14] Wei Meng et al. “Integrated multi-agent system framework: decentralised search, tasking and tracking”. In: *IET Control Theory & Applications* 9.3 (2014), pp. 493–502.
- [MH03] Scott McKenney and Robert Hunter. *Method for managing resource assets for emergency situations*. US Patent App. 10/038,572. July 2003.
- [MHL99] J-J Ch Meyer, Wiebe van der Hoek, and Bernd van Linder. “A logical approach to the dynamics of commitments”. In: *Artificial Intelligence* 113.1-2 (1999), pp. 1–40.
- [MST06] Janusz Marecki, Nathan Schurr, and Milind Tambe. “Agent-Based Simulations for Disaster Rescue Using the DEFACTO Coordination System.” In: *Emergent Information Technologies and Enabling Policies for Counter-Terrorism*. 2006, pp. 281–297.
- [Muk+17] Ayan Mukhopadhyay et al. “Prioritized Allocation of Emergency Responders based on a Continuous-Time Incident Prediction Model”. In: *Proceedings of the 16th Conference on Autonomous Agents and Multi-Agent Systems*. International Foundation for Autonomous Agents and Multiagent Systems. 2017, pp. 168–177.
- [Mye13] Roger B Myerson. *Game theory*. Harvard university press, 2013.
- [NSR00] Emma Norling, Liz Sonenberg, and Ralph Rönquist. “Enhancing multi-agent based simulation with human-like decision making strategies”. In: *International Workshop on Multi-Agent Systems and Agent-Based Simulation*. Springer. 2000, pp. 214–228.
- [OM98] Wilson W Orr and Raymond MP Miller. *Method for acquiring and presenting data relevant to an emergency incident*. US Patent 5,815,417. Sept. 1998.
- [OR94] Martin J Osborne and Ariel Rubinstein. *A course in game theory*. MIT press, 1994.

- [Pau+09] Ivan Pau et al. “Home e-health system integration in the Smart Home through a common media server”. In: *Proceedings of Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE. 2009, pp. 6171–6174.
- [Pet85] William J Petak. “Emergency management: A challenge for public administration”. In: *Public Administration Review* 45 (1985), pp. 3–7.
- [Pit+08] James Pita et al. “Deployed ARMOR protection: the application of a game theoretic model for security at the Los Angeles International Airport”. In: *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems: Industrial Track*. International Foundation for Autonomous Agents and Multiagent Systems. 2008, pp. 125–132.
- [Pit+11] James Pita et al. “GUARDS: game theoretic security allocation on a national scale”. In: *The 10th International Conference on Autonomous Agents and Multiagent Systems*. International Foundation for Autonomous Agents and Multiagent Systems. 2011, pp. 37–44.
- [Pon+10] Sameera Ponda et al. “Decentralized planning for complex missions with dynamic communication constraints”. In: *American Control Conference (ACC)*. IEEE. 2010, pp. 3998–4003.
- [PT03] Klaus-Dieter Pillekamp and Manfred Tasto. *Universal mobile telecommunications system*. US Patent 6,535,731. Mar. 2003.
- [R+95] Anand S Rao, Michael P Georgeff, et al. “BDI agents: From theory to practice.” In: *Proceedings Of the First International Conference on Multi-Agent Systems*. 1995, pp. 312–319.
- [Ram+08] Sarvapali D Ramchurn et al. “Agent-based coordination technologies in disaster management”. In: *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems: Demo Psapers*. International Foundation for Autonomous Agents and Multiagent Systems. 2008, pp. 1651–1652.
- [RJB04] James Rumbaugh, Ivar Jacobson, and Grady Booch. *The Unified Modeling Language Reference Manual*. Pearson Higher Education, 2004.
- [Rub12] Claire B Rubin. *Emergency Management: The American Experience 1900-2010*. CRC Press, 2012.
- [Saa06] Richard Saalsaa. *Method and system for integrating a computer aided dispatch system with an emergency medical dispatch protocol*. US Patent 7,106,835. Sept. 2006.

- [Sag04] Scott D Sagan. “Learning from normal accidents”. In: *Organization & Environment* 17.1 (2004), pp. 15–19.
- [SCL05] S Suárez, J Collins, and B López. “Improving rescue operations in disasters: approaches about task allocation and re-scheduling”. In: *24rd Annual Workshop of the UK Planning and Scheduling Special Interest Group (PlanSIG)*. 2005, pp. 66–75.
- [She10] Jiuh-Biing Sheu. “Dynamic relief-demand management for emergency logistics operations under large-scale disasters”. In: *Transportation Research Part E: Logistics and Transportation Review* 46.1 (2010), pp. 1–17.
- [ŠHJ13] Dragutin Šćap, Matija Hoić, and Andrej Jokić. “Determination Of the Pareto Frontier for Multiobjective Optimisation Problem”. In: *Transactions of FAMENA* 37.2 (2013), pp. 15–28.
- [SL06] Silvia Suárez and Beatriz López. “Reverse Combinatorial Auctions for Resource Allocation in the Rescue Scenario”. In: *Proceedings of the 16th International Conference on Automated Planning and Scheduling*. 2006, pp. 62–65.
- [SL16] Inkyung Sung and Taesik Lee. “Optimal allocation of emergency medical resources in a mass casualty incident: Patient prioritization by column generation”. In: *European Journal of Operational Research* 252.2 (2016), pp. 623–634.
- [Smi80] R Smith. “The contract net protocol: Highlevel communication and control in a distributed problem solver”. In: *IEEE Transactions on Computers* C-29.12 (1980), pp. 1104–1113.
- [Sno02] David Snowden. “Complex acts of knowing: paradox and descriptive self-awareness”. In: *Journal of knowledge management* 6.2 (2002), pp. 100–111.
- [Ste90] Luc Steels. “Cooperation between distributed agents through self-organisation”. In: *Proceedings of Intelligent Robots and Systems '90*. IEEE. 1990, pp. 8–14.
- [Su+14] Xing Su et al. “A dynamic coordination approach for task allocation in disaster environments under spatial and communicational constraints”. In: *AAAI Workshop on Multiagent Interaction without Prior Coordination (MIPC 2014)*. 2014.
- [Su+16] Zhaopin Su et al. “Multiple emergency resource allocation for concurrent incidents in natural disasters”. In: *International Journal of Disaster Risk Reduction* 17 (2016), pp. 199–212.

- [Swe94] Arthur J Swersey. “The deployment of police, fire, and emergency medical units”. In: *Handbooks in Operations Research and Management Science* 6 (1994), pp. 151–200.
- [Tar04] Xavier Tari. *Centralised system and method for dispensing services to automatic sales terminals*. US Patent App. 10/343,559. Jan. 2004.
- [Tsa+10a] Jason Tsai et al. “How to protect a city: Strategic security placement in graph-based domains”. In: *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems*. International Foundation for Autonomous Agents and Multiagent Systems. 2010, pp. 1453–1454.
- [Tsa+10b] Jason Tsai et al. “Urban security: Game-theoretic resource allocation in networked physical domains”. In: *National Conference on Artificial Intelligence*. 2010.
- [Tur+04] Murray Turoff et al. “The design of a dynamic emergency response management information system (DERMIS)”. In: *Journal of Information Technology Theory and Application* 5.4 (2004), p. 1.
- [Tur02] Murray Turoff. “Past and future emergency response information systems”. In: *Communications of the ACM* 45.4 (2002), pp. 29–32.
- [Ven+08] Celine Vens et al. “Decision trees for hierarchical multi-label classification”. In: *Machine Learning* 73.2 (2008), pp. 185–214.
- [Wau99] William L Waugh. *Living with hazards, dealing with disasters: An introduction to emergency management*. ME Sharpe, 1999.
- [Wei+14] Sean Shao Wei Lam et al. “Reducing ambulance response times using discrete event simulation”. In: *Prehospital Emergency Care* 18.2 (2014), pp. 207–216.
- [Wex+14] Felix Wex et al. “Emergency response in natural disaster management: Allocation and scheduling of rescue units”. In: *European Journal of Operational Research* 235.3 (2014), pp. 697–708.
- [Wha95] Jack Whalen. “The Social and Interactional Dimensions of Human-computer Interfaces”. In: Cambridge University Press, 1995. Chap. Expert Systems Versus Systems for Experts: Computer-aided Dispatch As a Support System in Real-world Environments, pp. 161–183.
- [WHM15] Michael J Widener, Mark W Horner, and Kunlei Ma. “Positioning Disaster Relief Teams Given Dynamic Service Demand: A Hybrid Agent-Based and Spatial Optimization Approach”. In: *Transactions in GIS* 19.2 (2015), pp. 279–295.

- [WJ95] Michael Wooldridge and Nicholas R Jennings. “Intelligent agents: theory and practice”. In: *The Knowledge Engineering Review* 10.02 (1995), pp. 115–152.
- [Woo97] Michael Wooldridge. “Agent-based software engineering”. In: *IEEE Proceedings - Software Engineering* 144.1 (1997), pp. 26–37.
- [WWS07] Su-Sheng Wang, Yan Wang, and Jian Sun. “An optimized emergency resources allocation algorithm for large-scale public emergency”. In: *2007 International Conference on Machine Learning and Cybernetics*. 2007, pp. 119–123.
- [XW01] Lai Xu and Hans Weigand. “The evolution of the contract net protocol”. In: *Advances in Web-Age Information Management*. Springer, 2001, pp. 257–264.
- [Yan+13] Zhaosheng Yang et al. “Multiobjective Model for Emergency Resources Allocation”. In: *Mathematical Problems in Engineering* 2013 (2013).
- [Yao+10] Junfeng Yao et al. “Path planning for virtual human motion using improved A\* star algorithm”. In: *Seventh International Conference on Information Technology: New Generations*. IEEE. 2010, pp. 1154–1158.
- [YH95] Michael Youssefmir and Bernardo A Huberman. “Resource Contention in Multiagent Systems.” In: *Proceedings of the First International Conference on Multiagent Systems (ICMAS)*. 1995, pp. 398–405.
- [ZA08] Konstantinos G Zografos and Konstantinos N Androutsopoulos. “A decision support system for integrated hazardous materials routing and emergency response decisions”. In: *Transportation Research Part C: Emerging Technologies* 16.6 (2008), pp. 684–703.
- [Zay+13] Gabriel Zayas-Cabán et al. “Emergency medical service allocation in response to large-scale events”. In: *IEEE Transactions on Healthcare Systems Engineering* 3.1 (2013), pp. 57–68.
- [ZC09] W Zeng and RL Church. “Finding shortest paths on real road networks: the case for A\*”. In: *International Journal of Geographical Information Science* 23.4 (2009), pp. 531–543.
- [ZLL12] Jiang-Hua Zhang, Jin Li, and Zhi-Ping Liu. “Multiple-resource and multiple-depot emergency response problem considering secondary disasters”. In: *Expert Systems with Applications* 39.12 (2012), pp. 11066–11071.

- [ZO04] Franco Zambonelli and Andrea Omicini. “Challenges and research directions in agent-oriented software engineering”. In: *Autonomous Agents and Multi-Agent Systems* 9.3 (2004), pp. 253–283.

# Appendix A

## An Agent-Based Emergency Resource Deployment Simulator

During this PhD study, an agent-based emergency resource deployment simulator was designed and implemented to evaluate the proposed resource deployment models, algorithms and mechanisms introduced in Chapters 3, 4 and Chapter 5. The proposed simulator could also provide the demonstration of the resource deployment process to emergency incidents.

### A.1 Implementation Features of the Simulator

The proposed agent-based resource deployment simulator was implemented as a web-based system with the Model-View-Controller (MVC) architecture [LR01], which could be accessed from any computer that has a web browser and the Internet connection.

With the MVC architecture design, the proposed simulator separated the presentation layer from the logic layer. By minimising the clutter, the simulator could be easily extended with new modules for broadening its applications in the future. Besides, in order to evaluate the resource deployment models, algorithms, and mechanisms, introduced in Chapters 3, 4 and Chapter 5 in the real-world environments, GoogleMaps technology was integrated into the View layer of the proposed simulator. The GoogleMap technology enabled the proposed simulator to search for the real-world information of local facility resources in any metropolitan region, such as



the location of fire stations, hospitals and police stations. The information of city routes and life traffics could also be acquired by the simulator to demonstrate the resource movements on the GoogleMaps.

In addition, Java Agent Development Framework (JADE) was used for the implementation of the agents in the Controller layer [BCG07]. The major reason for choosing JADE as the agent platform was that JADE could provide a dynamic runtime environment for an agent's registration and operation. This dynamic runtime environment was extremely important for simulating the resource deployment process for unpredictable emergency incidents with different variations in metropolitan regions.

Random Demo Setting Detail	
Event Content	Vehical Accident
Event Severity	Level 4
Task Number	3
Task Average Deadline	7 min
Resource Average Expenditure	44
Resource Average Velocity	62 km/h
Average Resource in Facility	3

Reset Start

(a) Random Setting

Advance Demo Setting Detail	
Event Severity [1 - 5]	Task Number
Task Average Deadline (min)	Resource Average Expenditure
Resource Average Speed (km/h)	Average Resource in Facility

Reset Start

(b) Advance Setting

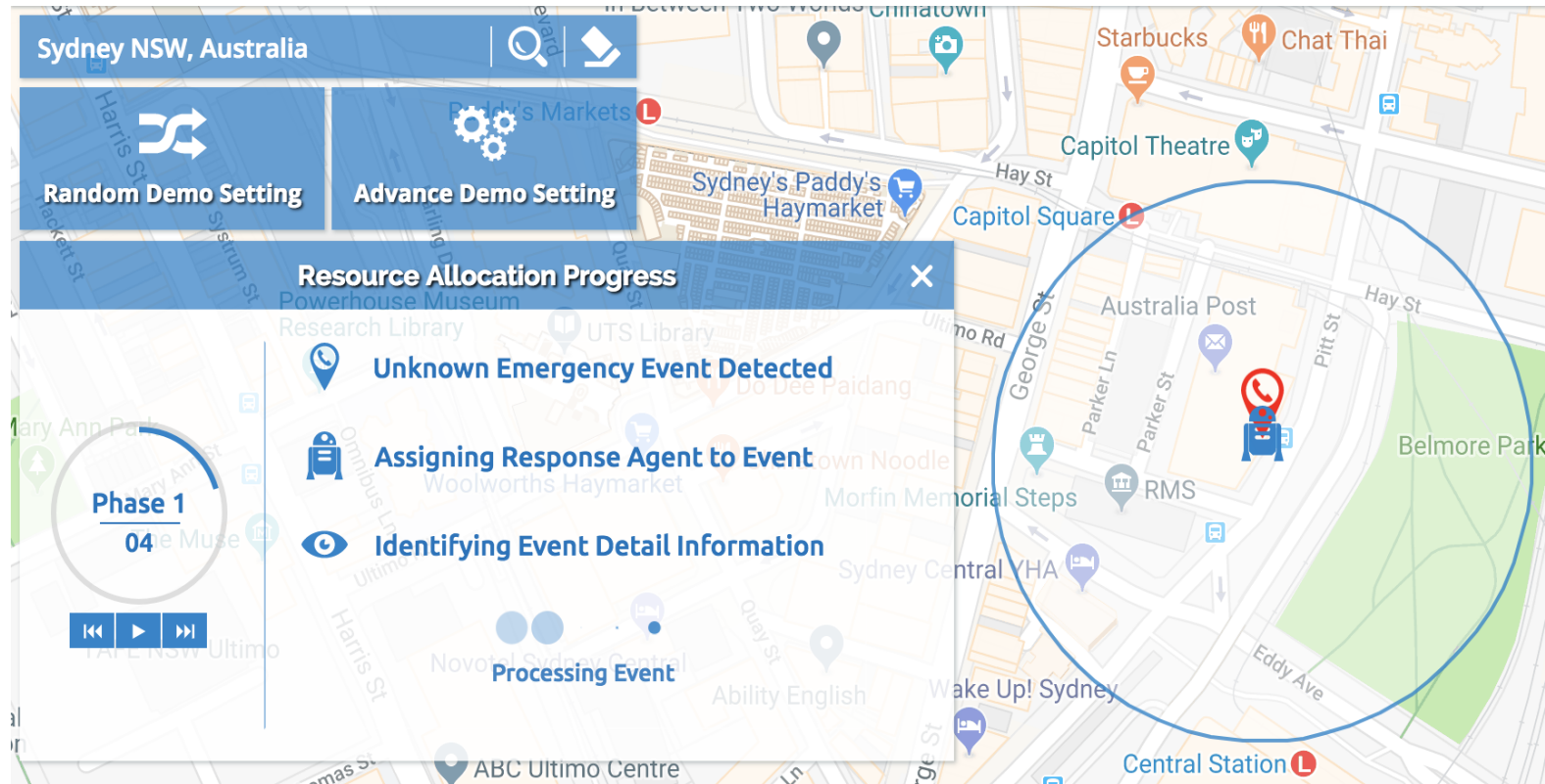
Figure A.1: Demo Setting

Furthermore, the proposed simulator could provide highly interactive demonstrations by using JQuery and Ajax Technologies [Gyö+09]. More precisely, the proposed simulator could allow users to place any number of emergency incidents in any city around the world. Users could choose to let the simulator to automatically generate random demo settings (i.e. Figure A.1(a)) or to manually configure the settings of emergency incidents (i.e. Figure A.1(b)). During a resource deployment progress, the proposed simulator used multiple animations to illustrate how agents interacted and coordinated with each other to make decisions, and how rescue resources were moved dynamically on the Google Maps with the effect of live traffic.

## A.2 Simulator Demonstration

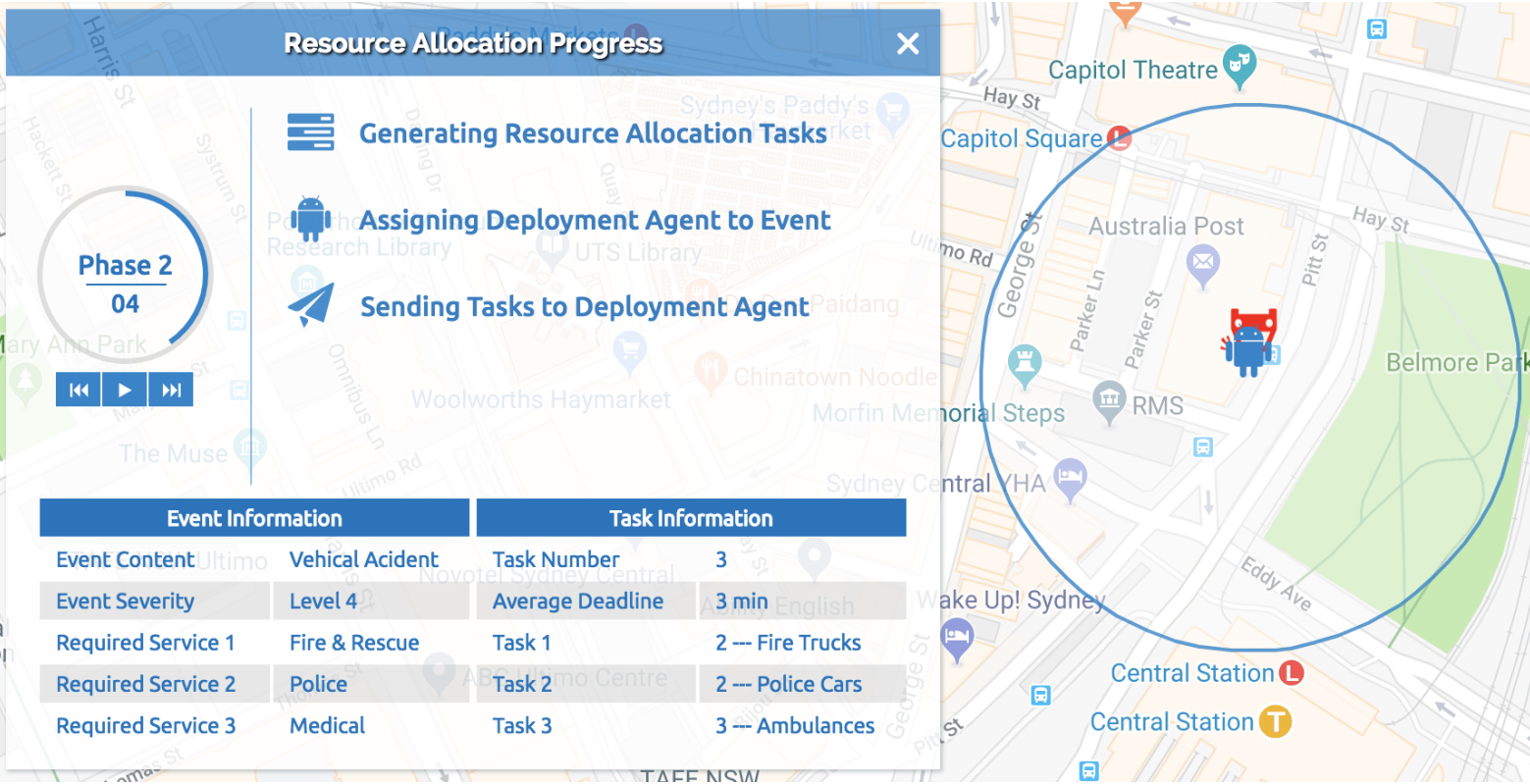
This section gives a demonstration of the detailed resource deployment process for a single emergency incident in the proposed agent-based simulator. Generally, the demonstration of the single incident resource deployment can be breakdown into four phases.

**Phase 1** (i.e. Figure A.2): when the information of an emergency incident is received by the proposed simulator, a processing thread will be automatically generated to handle the resource deployment of this incident. First, a response agent is assigned to this incident to analyse the incident information.



**Figure A.2:** Single Incident Resource Deployment — Phase 1

**Phase 2** (i.e. Figure A.3): after the response agent identifies the incident’s information, a set of resource deployment tasks will be generated based on the incident’s content, severity and required emergency services. These tasks will be passed to a deployment agent.



**Figure A.3:** Single Incident Resource Deployment — Phase 2

**Phase 3** (i.e. Figure A.4): the deployment agent will send task information to relevant emergency facilities near the incident location. Then, a facility agent, who received a resource deployment task, will propose a resource deployment plan based on the availability of its mobile resources and will send the plan back to the deployment agent.

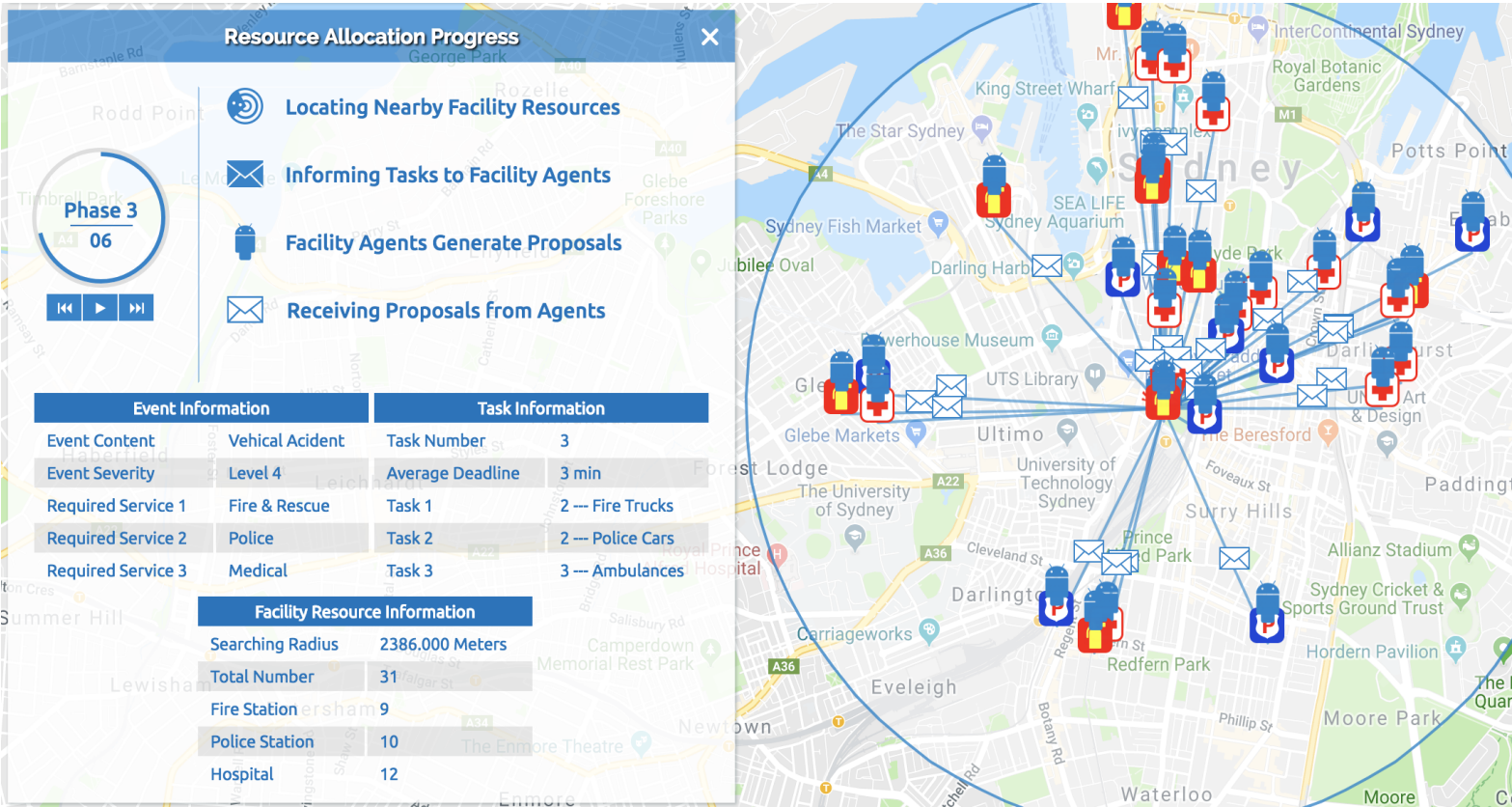
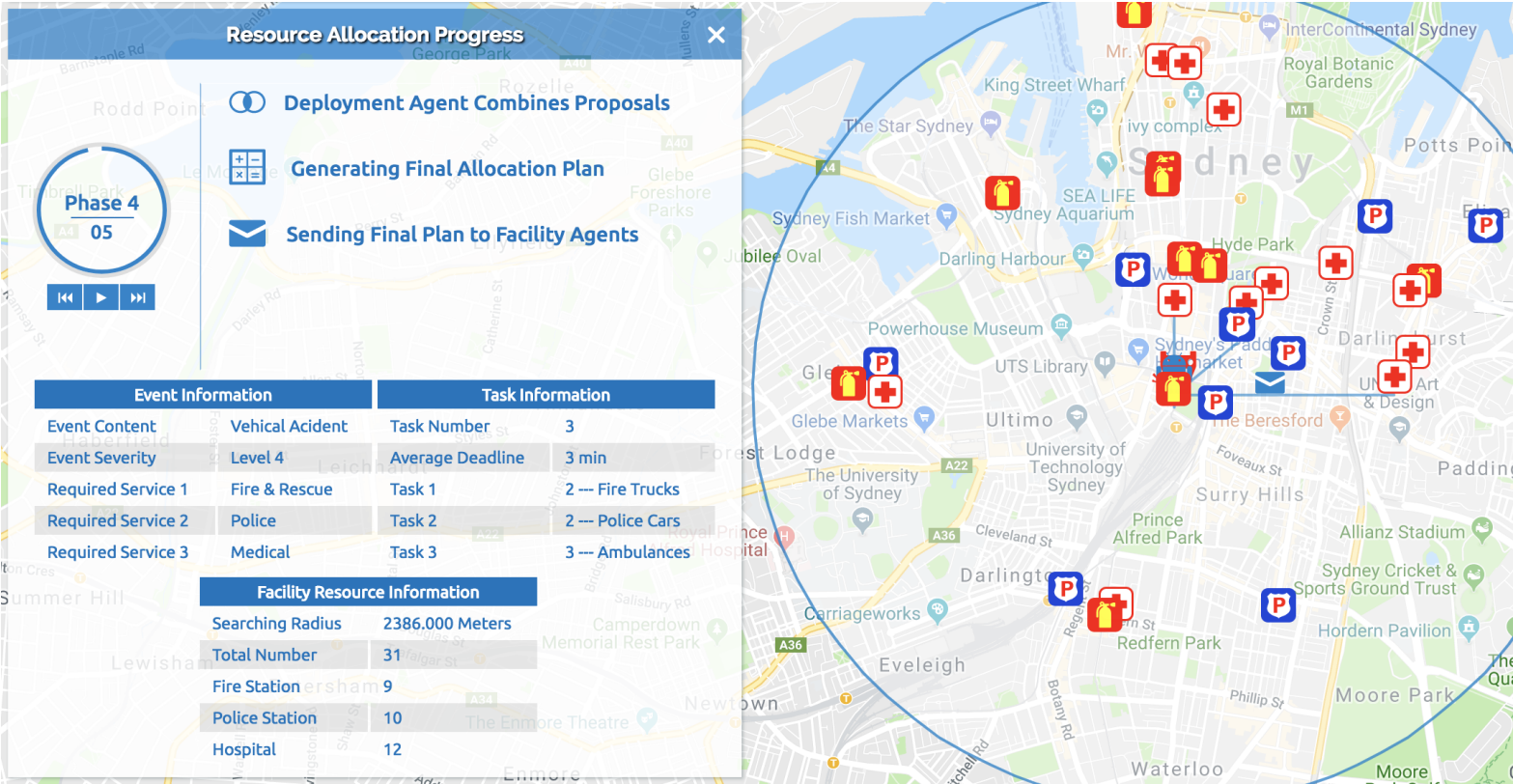


Figure A.4: Single Incident Resource Deployment — Phase 3



**Phase 4** (i.e. Figure A.5): after the deployment agent receives all resource deployment plans from selected facility agents, it will combine these plans and use the domain transportation theory to generate an optimal plan for each resource deployment task. Finally, each of the resource deployment plans will be sent to relevant emergency facilities, in which facility agents will inform mobile agents to execute the resource deployment requests.



**Figure A.5:** Single Incident Resource Deployment — Phase 4

At the end of the resource deployment progress, the proposed simulator will generate a detail resource deployment report, which includes the total cost of the resource deployment, average deployment time and so on (see Figure A.6 for detail).

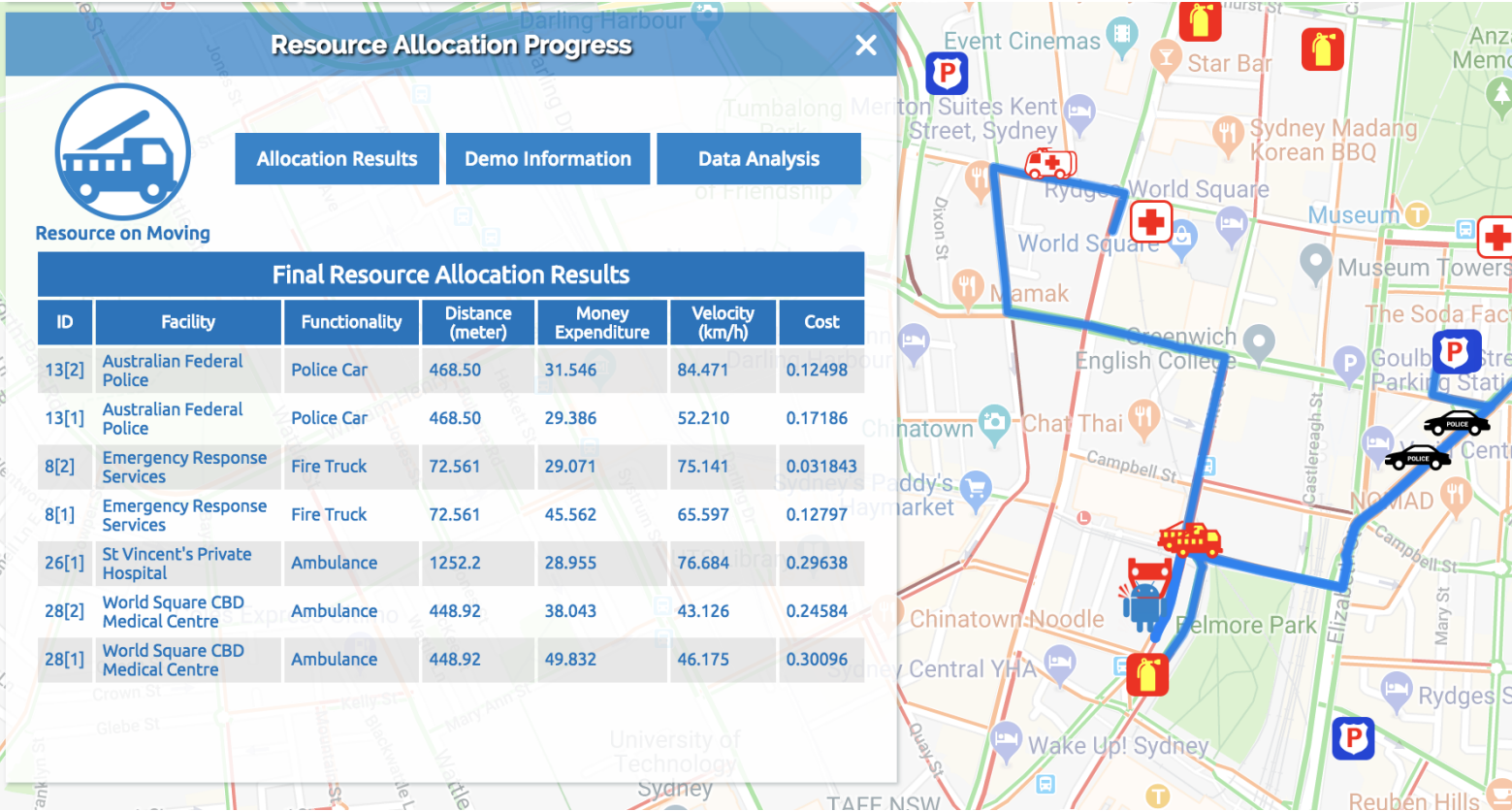


Figure A.6: Resource Deployment Request Execution